Bisimulations for Delimited-Control Operators

Dariusz Biernacki, Sergueï Lenglet, Piotr Polesiuk

Bisimulations for Delimited-Control Operators

Dariusz Biernacki, Sergueï Lenglet, Piotr Polesiuk

ISSN 0249-6399 ISRN INRIA/RR--9096--FR+ENG

RESEARCH REPORT
N° 9096
September 2017
Project-Teams Pareo

Bisimulations for Delimited-Control Operators

Dariusz Biernacki[*], Sergueï Lenglet[†], Piotr Polesiuk[*]

Project-Teams Pareo

Research Report n° 9096 — September 2017 — 107 pages

Abstract: We propose a survey of the behavioral theory of an untyped lambda-calculus extended with the delimited-control operators shift and reset. We define a contextual equivalence for this calculus, that we then aim to characterize with coinductively defined relations, called *bisimilarities*. We study different styles of bisimilarities (namely applicative, normal-form, and environmental), and we give several examples to illustrate their respective strengths and weaknesses. We also discuss how to extend this work to other delimited-control operators.

Key-words: behavioral equivalences, λ-calculus, delimited continuation, control operators

[*] University of Wrocław
[†] Université de Lorraine

RESEARCH CENTRE
NANCY – GRAND EST

615 rue du Jardin Botanique
CS20101
54603 Villers-lès-Nancy Cedex

Bisimulations pour les opérateurs de contrôle délimité

Résumé : Nous proposons un panorama de la théorie comportementale d'un lambda-calcul non typé étendu avec les opérateurs de contrôle délimité shift et reset. Nous définissons une équivalence contextuelle, que nous cherchons à caractériser avec des *bisimilarités* définies coinductivement. Nous étudions plusieurs styles de bisimilarité (applicative, de forme normale, environnementale) et donnons plusieurs exemples pour illustrer leurs forces et faiblesses respectives. Nous discutons également de comment étendre ce travail à d'autres opérateurs de contrôle délimité.

Mots-clés : Équivalences comportementales, λ-calcul, continuation délimitée, opérateurs de contrôle

1. Introduction

Delimited-control operators. Control operators for delimited continuations enrich a programming language with the ability to delimit the current continuation, to capture such a delimited continuation, and to compose delimited continuations. Such operators have been originally proposed independently by Felleisen [24] and by Danvy and Filinski [18], with numerous variants designed subsequently [33, 63, 31, 23]. The applications of delimited-control operators range from non-deterministic programming [18, 43], partial evaluation [16], and normalization by evaluation [22] to concurrency [33], mobile code [81], linguistics [77], operating systems [41], and probabilistic programming [42]. Several variants of delimited-control operators are nowadays available in mainstream functional languages such as Haskell [23], Ocaml [40], Scala [70], Scheme [28], and SML [26].

The control operators shift and reset [18] were designed to account for the traditional model of non-deterministic programming based on success and failure continuations, and their semantics as well as pragmatics take advantage of an extended continuation-passing style (CPS), where the continuation of the computation is represented by the current delimited continuation and a meta-continuation. In his seminal articles [26, 27], Filinski showed that because the continuation monad can express any other monad, shift and reset can express any monadic effect in direct style (DS), which gives them a special position among all the control operators considered in the literature. In particular, the control operator $\mathsf{call/cc}$ known from Scheme and SML of New Jersey requires the presence of mutable state to obtain the expressive power of shift and reset.

Relying on the CPS translation to a pure language is helpful and inspiring when programming with shift and reset, but it is arguably more convenient to reason directly about the code with control operators. To facilitate such reasoning, Kameyama et al. devised direct-style axiomatizations for a number of delimited-control calculi [38, 37, 39] that are sound and complete with respect to the corresponding CPS translations. Numerous other results concerning equational reasoning in various calculi for delimited continuations [71, 2, 32, 59] show that it has been a topic of active research.

While the CPS-based equational theories are a natural consequence of the denotational or translational semantics of control operators such as shift and reset, they are not strong enough to verify the equivalences of programs that have unrelated images through the CPS translation, but that operationally cannot be distinguished (e.g., take two different fixed-point combinators). In order to build a stronger theory of program equivalence for delimited control, we turn to the operational foundations of shift and reset [7], and consider operationally-phrased criteria for program equivalence.

Behavioral equivalences. Because of the complex nature of control effects, it can be difficult to determine if two programs that use shift and reset are equivalent (i.e., behave in the same way) or not. *Contextual equivalence* [64] is widely considered as the most natural equivalence on terms in languages based on the λ-calculus. The intuition behind this relation is that two programs are equivalent if replacing one by the other in a bigger program does not change the behavior of this bigger program. The behavior of a program has to be made formal by defining the *observable actions* we want to take into account for the calculus we consider. It can be, e.g., inputs and outputs for communicating systems [75], memory reads and writes, etc. For the plain λ-calculus [1], it is usually whether the term terminates or not. The "bigger program" can be seen as a *context* (a term with a hole) and, therefore, two terms t_0 and t_1 are contextually equivalent if we cannot tell them apart when executed within any context C, i.e., if $C[t_0]$ and $C[t_1]$ produce the same observable actions.

The latter quantification over contexts C makes contextual equivalence hard to use in practice to prove that two given terms are equivalent. As a result, one usually looks for more tractable alternatives to contextual equivalence, such as logical relations (see, e.g., [68]), axiomatizations (see, e.g., [48]), or *bisimulations*. A bisimulation relates two terms t_0 and t_1 by asking them to mimic each other in a coinductive way, e.g., if t_0 reduces to a term t'_0, then t_1 has to reduce to a term t'_1 so that t'_0 and t'_1 are still in the bisimulation, and conversely for the reductions of t_1. An equivalence on terms, called *bisimilarity* can be derived from a notion of bisimulation: two terms are bisimilar if there exists a bisimulation which relates them. Finding an appropriate notion of bisimulation consists in finding the conditions on which two terms are related, so that the resulting notion of bisimilarity is *sound* and *complete* w.r.t. contextual equivalence, (i.e., it is included in and it contains contextual equivalence, respectively).

Different styles of bisimulations have been proposed for calculi similar to the λ-calculus. For example, *applicative* bisimilarity [1] relates terms by reducing them to values (if possible), and the resulting values have to be themselves applicative bisimilar when applied to an arbitrary argument. As we can see, applicative bisimilarity still contains some quantification over arguments to compare values, but is nevertheless easier to use than contextual equivalence because of its coinductive nature, and also because we do not have to consider all forms of contexts. Applicative bisimilarity is usually sound and complete w.r.t. contextual equivalence, at least for deterministic languages such as the plain λ-calculus [1].

In contrast with applicative bisimilarity, *normal-form* bisimilarity [52] (also called *open* bisimilarity in [72]) does not contain any quantification over arguments or contexts in its definition. The principle is to reduce the compared terms to normal forms (if possible), and then to decompose the resulting normal forms into sub-components that have to be themselves bisimilar. Unlike applicative bisimilarity, normal-form bisimilarity is usually not complete, i.e., there exist contextually equivalent terms that are not normal-form bisimilar. But because of the lack of quantification over contexts, proving that two terms are normal-form bisimilar is usually quite simple, and the proofs can be further simplified with the help of *up-to techniques*. The idea behind up-to techniques is to define relations that are not exactly bisimulations but are included in bisimulations. Finding an up-to relation equating two given terms is usually simpler than finding a regular bisimulation relating these terms.

Finally, *environmental bisimilarity* [74] is quite similar to applicative bisimilarity, as it compares terms by reducing them to values, and then requires the resulting values to be bisimilar when applied to some arguments. However, the arguments are no longer arbitrary, but built using an *environment*, which represents the knowledge accumulated so far by an outside observer on the tested terms. Like applicative bisimilarity, environmental bisimilarity is usually sound and complete, but it also allows for up-to techniques (like normal-form bisimilarity) to simplify its equivalence proofs. In contrast, the definition of useful up-to techniques for applicative bisimilarity remains an open problem.

This work. In this article, we propose a survey of the behavioral theory of a λ-calculus extended with the operators shift and reset, called $\lambda_{\mathcal{S}}$. In previous works, we defined applicative [11], normal-form [12], and environmental [13] bisimilarities for this calculus. We present here these results in a systematic and uniform way, with examples allowing for comparisons between the different styles of bisimulation. In particular, we compare bisimilarities to Kameyama and Hasegawa's direct style axiomatization of $\lambda_{\mathcal{S}}$ [38], and we use these axioms as examples throughout the paper. We consider two semantics for $\lambda_{\mathcal{S}}$, one that is faithful to its defining CPS translation, where terms are executed within an outermost reset (we call it the "original semantics"), and another one where this requirement is lifted (we call it the "relaxed semantics"). Finally, we discuss how this work can be extended to other delimited-control operators.

Structure of the article. Section 2 presents the syntax and semantics of the calculus $\lambda_{\mathcal{S}}$ with shift and reset that we use in this paper. In this section, we also recall the definition of CPS equivalence, a CPS-based equivalence between terms, and its axiomatization. Section 3 discusses the definition of a contextual equivalence for $\lambda_{\mathcal{S}}$, and its relationship with CPS equivalence. We look for (at least sound) alternatives of this contextual equivalence by considering several styles of bisimilarities: normal-form in Section 4, applicative in Section 5, and environmental in Section 6. Section 7 discusses the possible extensions of our work to other semantics and other calculi with delimited control, and Section 8 concludes this paper. The appendices contain the proofs too long to be included in the body of the paper.

We discuss related work along the way in the relevant sections, e.g., related work on normal-form bisimilarities for control operators is discussed at the beginning of Section 4. Section 4 develops on results presented in [12], except for the part about the original semantics (Section 4.6), which is new. Section 5 expands on results in [11], and Section 6 on results in [13], except that the definition of environmental bisimulation for the relaxed semantics has been changed.

Notations and basic definitions. We define here some notations frequently used throughout the paper. We write $\stackrel{\mathsf{def}}{=}$ for a defining equality, i.e., $m \stackrel{\mathsf{def}}{=} e$ means that m is defined as the expression e. Given a metavariable m, we write $\overrightarrow{m}$ for a sequence of entities denoted by m. Given a binary relation $\mathcal{R}$, we write $m \mathrel{\mathcal{R}} m'$

for $(m, m') \in \mathcal{R}$, $\mathcal{R}^{-1}$ for its inverse, defined as $\mathcal{R}^{-1} \stackrel{\mathsf{def}}{=} \{(m', m) \mid m \mathrel{\mathcal{R}} m'\}$, and $\mathcal{R}^*$ for its transitive and reflexive closure, defined as $\mathcal{R}^* \stackrel{\mathsf{def}}{=} \{(m, m') \mid \exists m_1, \ldots, m_k, k \geq 0 \wedge m = m_0 \wedge m_k = m' \wedge \forall 0 \leq i < k, m_i \mathrel{\mathcal{R}} m_{i+1}\}$. Further, given two binary relations $\mathcal{R}$ and $\mathcal{S}$ we use juxtaposition $\mathcal{RS}$ for their composition, defined as $\mathcal{RS} = \{(m, m') \mid \exists m'', m \mathrel{\mathcal{R}} m'' \wedge m'' \mathrel{\mathcal{S}} m'\}$. Finally, a relation $\mathcal{R}$ is *compatible* if it is preserved by all the operators of the language, e.g., $t_0 \mathrel{\mathcal{R}} t_1$ implies $\lambda x.t_0 \mathrel{\mathcal{R}} \lambda x.t_1$; a relation is a *congruence* if it is a compatible equivalence relation.

2. The Calculus

In this section, we present the syntax, reduction semantics, and CPS equivalence for the language $\lambda_{\mathcal{S}}$ studied throughout this article.

2.1. Syntax

The language $\lambda_{\mathcal{S}}$ extends the call-by-value λ-calculus with the delimited-control operators shift and reset [18]. We assume we have a set of term variables, ranged over by x, y, z, and k. We use the metavariable k for shift-bound variables representing a continuation, while x, y, and z stand for the usual lambda-bound variables representing any values; we believe such a distinction helps to understand examples and reduction rules.

The syntax of terms ($\mathcal{T}$) and values ($\mathcal{V}$) is given by the following grammars:

$$\begin{array}{llcl} \text{Terms:} & t & ::= & v \mid t\,t \mid \mathcal{S}k.t \mid \langle t \rangle \\ \text{Values:} & v & ::= & x \mid \lambda x.t \end{array}$$

The operator shift ($\mathcal{S}k.t$) is a capture operator, the extent of which is determined by the delimiter reset ($\langle \cdot \rangle$). A λ-abstraction $\lambda x.t$ binds x in t and a shift construct $\mathcal{S}k.t$ binds k in t; terms are equated up to α-conversion of their bound variables. The set of free variables of t is written $\mathsf{fv}(t)$; a term t is *closed* if $\mathsf{fv}(t) = \emptyset$. The set of closed terms (values) is noted $\mathcal{T}_c$ ($\mathcal{V}_c$, respectively).

We distinguish several kinds of contexts, represented outside-in, as follows:

$$\begin{array}{llcl} \text{Pure contexts:} & E & ::= & \square \mid v\,E \mid E\,t \\ \text{Evaluation contexts:} & F & ::= & \square \mid v\,F \mid F\,t \mid \langle F \rangle \\ \text{Contexts:} & C & ::= & \square \mid \lambda x.C \mid t\,C \mid C\,t \mid \mathcal{S}k.C \mid \langle C \rangle \end{array}$$

Regular contexts are ranged over by C. The pure evaluation contexts ($\mathcal{PC}$) (abbreviated as pure contexts),[1] ranged over by E, represent delimited continuations and can be captured by the shift operator. The call-by-value evaluation contexts, ranged over by F, represent arbitrary continuations and encode the chosen reduction strategy. Filling a context C (E, F) with a term t produces a

[1] This terminology comes from Kameyama (e.g., in [38]); note that we use the metavariables of [7] for evaluation contexts, which are reversed compared to [38].

term, written $C[t]$ ($E[t]$, $F[t]$, respectively); the free variables of t may be captured in the process. We extend the notion of free variables to contexts (with $\mathsf{fv}(\Box) = \emptyset$), and we say a context C (E, F) is *closed* if $\mathsf{fv}(C) = \emptyset$ ($\mathsf{fv}(E) = \emptyset$, $\mathsf{fv}(F) = \emptyset$, respectively). The set of closed pure contexts is noted $\mathcal{PC}_c$. In any definitions or proofs, we say a variable is *fresh* if it does not occur free in the terms or contexts under consideration.

2.2. Reduction Semantics

The call-by-value left-to-right reduction semantics of $\lambda_{\mathcal{S}}$ is defined as follows, where $t\{v/x\}$ is the usual capture-avoiding substitution of v for x in t:

$$\begin{array}{rcll} F[(\lambda x.t)\, v] & \rightarrow_{\mathrm{v}} & F[t\{v/x\}] & (\beta_v) \\ F[\langle E[\mathcal{S}k.t]\rangle] & \rightarrow_{\mathrm{v}} & F[\langle t\{\lambda x.\langle E[x]\rangle/k\}\rangle] \text{ with } x \notin \mathsf{fv}(E) & (\mathit{shift}) \\ F[\langle v\rangle] & \rightarrow_{\mathrm{v}} & F[v] & (\mathit{reset}) \end{array}$$

The term $(\lambda x.t)\, v$ is the usual call-by-value redex for β-reduction (rule (β_v)). The operator $\mathcal{S}k.t$ captures its surrounding context E up to the dynamically nearest enclosing reset, and substitutes $\lambda x.\langle E[x]\rangle$ for k in t (rule (*shift*)). If a reset is enclosing a value, then it has no purpose as a delimiter for a potential capture, and it can be safely removed (rule (*reset*)). All these reductions may occur within a metalevel context F. The chosen call-by-value evaluation strategy is encoded in the grammar of the evaluation contexts. Furthermore, the reduction relation $\rightarrow_{\mathrm{v}}$ is compatible with evaluation contexts F, i.e., $F[t] \rightarrow_{\mathrm{v}} F[t']$ whenever $t \rightarrow_{\mathrm{v}} t'$. We write $t \rightarrow_{\mathrm{v}}$ when there is a t' such that $t \rightarrow_{\mathrm{v}} t'$ and we write $t \not\rightarrow_{\mathrm{v}}$ when no such t' exists.

Example 2.1. Let $i \stackrel{\mathsf{def}}{=} \lambda x.x$ and $\omega \stackrel{\mathsf{def}}{=} \lambda x.x\,x$. We present the sequence of reductions initiated by $\langle((\mathcal{S}k_1.i\,(k_1\,i))\,\mathcal{S}k_2.\omega)\,(\omega\,\omega)\rangle$. The term $\mathcal{S}k_1.i\,(k_1\,i)$ is within the pure context $E \stackrel{\mathsf{def}}{=} (\Box\,\mathcal{S}k_2.\omega)\,(\omega\,\omega)$, enclosed in a delimiter $\langle\cdot\rangle$, so E is captured according to rule (*shift*):

$$\langle((\mathcal{S}k_1.i\,(k_1\,i))\,\mathcal{S}k_2.\omega)\,(\omega\,\omega)\rangle \rightarrow_{\mathrm{v}} \langle i\,((\lambda x.\langle(x\,\mathcal{S}k_2.\omega)\,(\omega\,\omega)\rangle)\,i)\rangle$$

The role of reset in $\lambda x.\langle E[x]\rangle$ is more clear after reduction of the β_v-redex $(\lambda x.\langle E[x]\rangle)\,i$:

$$\langle i\,((\lambda x.\langle(x\,\mathcal{S}k_2.\omega)\,(\omega\,\omega)\rangle)\,i)\rangle \rightarrow_{\mathrm{v}} \langle i\,\langle(i\,\mathcal{S}k_2.\omega)\,(\omega\,\omega)\rangle\rangle$$

When the captured context E is reactivated, it is not *merged* with the context $i\,\Box$, but *composed* thanks to the reset enclosing E. As a result, the capture triggered by $\mathcal{S}k_2.\omega$ leaves the term i outside the first enclosing reset intact:

$$\langle i\,\langle(i\,\mathcal{S}k_2.\omega)\,(\omega\,\omega)\rangle\rangle \rightarrow_{\mathrm{v}} \langle i\,\langle\omega\rangle\rangle$$

Because k_2 does not occur in ω, the context $(i\,\Box)\,(\omega\,\omega)$ is discarded when captured by $\mathcal{S}k_2.\omega$. Finally, we remove the useless delimiter $\langle i\,\langle\omega\rangle\rangle \rightarrow_{\mathrm{v}} \langle i\,\omega\rangle$ with rule (*reset*), and we then β_v-reduce and remove the last delimiter $\langle i\,\omega\rangle \rightarrow_{\mathrm{v}} \langle\omega\rangle \rightarrow_{\mathrm{v}} \omega$. Note that while the reduction strategy is call-by-value, some function arguments are not evaluated, like the non-terminating term $\omega\,\omega$ in this example.

Example 2.2 (fixed-point combinators). We recall the definition of Turing's and Curry's fixed-point combinators. Let $\theta \stackrel{\mathsf{def}}{=} \lambda xy.y\ (\lambda z.x\ x\ y\ z)$ and $\delta_x \stackrel{\mathsf{def}}{=} \lambda y.x(\lambda z.yyz)$; then $\Theta \stackrel{\mathsf{def}}{=} \theta\theta$ is Turing's call-by-value fixed-point combinator, and $\Delta \stackrel{\mathsf{def}}{=} \lambda x.\delta_x\ \delta_x$ is Curry's call-by-value fixed-point combinator. In [17], the authors propose variants of these combinators using shift and reset. They write Turing's combinator as $\langle\theta\ \mathcal{S}k.k\ k\rangle$ and Curry's combinator as $\lambda x.\langle\delta_x\ \mathcal{S}k.k\ k\rangle$. For an example, the following reduction sequence demonstrates the behavior of the former:

$$\langle\theta\ \mathcal{S}k.k\ k\rangle \rightarrow_{\mathrm{v}} \langle(\lambda x.\langle\theta\ x\rangle)\ (\lambda x.\langle\theta\ x\rangle)\rangle \rightarrow_{\mathrm{v}}^{*} \lambda y.y\ (\lambda z.(\lambda x.\langle\theta\ x\rangle)\ (\lambda x.\langle\theta\ x\rangle)\ y\ z)$$

We use the combinators and their delimited-control variants as examples throughout the paper, and, in particular, we study the equivalences between them in Example 4.4.

Remark 2.3. The context capture can also be written using local reduction rules [24], where the context is consumed piece by piece. We discuss these reduction rules and their consequences on the results of this article in Section 7.1.

There exist terms which are not values and which cannot be reduced any further; these are called *stuck terms*.

Definition 2.4. A term t is stuck if t is not a value and $t \not\rightarrow_{\mathrm{v}}$.

For example, the term $E[\mathcal{S}k.t]$ is stuck because there is no enclosing reset; the capture of E by the shift operator cannot be triggered. In fact, stuck terms are easy to characterize.

Proposition 2.5. *A term t is stuck iff*

- $t = E[\mathcal{S}k.t']$ *for some* E, k, *and* t', *or*
- $t = F[x\ v]$ *for some* F, x, *and* v.

We call *control stuck terms* the terms of the form $E[\mathcal{S}k.t]$ and *open stuck terms* the terms of the form $F[x\ v]$.

Definition 2.6. A term t is a normal form, if t is a value or a stuck term.

We call *redexes* (ranged over by r) terms of the form $(\lambda x.t)\ v$, $\langle E[\mathcal{S}k.t]\rangle$, and $\langle v\rangle$. Thanks to the following unique-decomposition property, the reduction relation $\rightarrow_{\mathrm{v}}$ is deterministic.

Proposition 2.7. *For all terms t, either t is a normal form, or there exist a unique redex r and a unique context F such that $t = F[r]$.*

Finally, we define the evaluation relation of $\lambda_{\mathcal{S}}$ as follows.

Definition 2.8. We write $t \Downarrow_{\mathrm{v}} t'$ if $t \rightarrow_{\mathrm{v}}^{*} t'$ and t' is a normal form.

If a term t admits an infinite reduction sequence, we say it *diverges*, written $t \Uparrow_{\mathrm{v}}$. As an example of such a term, we use extensively $\Omega \stackrel{\mathsf{def}}{=} (\lambda x.x\, x)\, (\lambda x.x\, x)$.

In the rest of the paper, we use the following results on the reduction (or evaluation) of terms. First, a control stuck term cannot be obtained from a term of the form $\langle t \rangle$.

Proposition 2.9. *If $\langle t \rangle \Downarrow_{\mathrm{v}} t'$ then t' is a value or an open stuck term of the form $\langle F[x\, v] \rangle$. (If t is closed then t' can only be a closed value.)*

Proof. By induction on the number of reduction steps in $\langle t \rangle \Downarrow_{\mathrm{v}} t'$.

We then show that reduction is preserved by substitution.

Proposition 2.10. *If $t \rightarrow_{\mathrm{v}} t'$, then $t\{v/x\} \rightarrow_{\mathrm{v}} t'\{v/x\}$.*

Proof. By case analysis on $t \rightarrow_{\mathrm{v}} t'$.

2.3. The original reduction semantics

Let us notice that the reduction semantics we have introduced does not require terms to be evaluated within a top-level reset—a requirement that is commonly relaxed in practical implementations of shift and reset [23, 26], but also in some other studies of these operators [5, 37]. This is in contrast to the original reduction semantics for shift and reset [7] that has been obtained from the 2-layered continuation-passing-style (CPS) semantics [18], discussed in Section 2.4. A consequence of the correspondence with the CPS-based semantics is that terms in the original reduction semantics are treated as complete programs and are decomposed into triples consisting of a subterm (a value or a redex), a delimited context, and a meta-context (a list of delimited contexts), resembling abstract machine configurations. Such a decomposition imposes the existence of an implicit top-level reset, hard-wired in the decomposition, surrounding any term to be evaluated.

The two semantics, therefore, differ in that in the original semantics there are no control stuck terms. However, it can be easily seen that operationally the difference is not essential—they are equivalent when it comes to terms of the form $\langle t \rangle$. In the rest of the article we call such terms *delimited terms* and we use the relaxed semantics when analyzing their behavior.

The top-level reset requirement, imposed by the original semantics, does not lend itself naturally to the notion of applicative bisimulation that we propose for the relaxed semantics in Section 5. We show, however, that the requirement can be successfully treated in the framework of normal-form and environmental bisimulations, presented in Sections 4 and 6.

2.4. CPS Equivalence

The operators shift and reset have been originally defined by a translation into continuation-passing style [18] that we present in Figure 1. Translated terms expect two continuations: the delimited continuation representing the

$$
\begin{array}{rcl}
\overline{x} & = & \lambda k_1 k_2.k_1\ x\ k_2 \\
\overline{\lambda x.t} & = & \lambda k_1 k_2.k_1\ (\lambda x.\overline{t})\ k_2 \\
\overline{t_0\ t_1} & = & \lambda k_1 k_2.\overline{t_0}\ (\lambda x_0 k_2'.\overline{t_1}\ (\lambda x_1 k_2''.x_0\ x_1\ k_1\ k_2'')\ k_2')\ k_2 \\
\overline{\langle t \rangle} & = & \lambda k_1 k_2.\overline{t}\ \gamma\ (\lambda x.k_1\ x\ k_2) \\
\overline{\mathcal{S}k.t} & = & \lambda k_1 k_2.\overline{t}\{(\lambda x_1 k_1' k_2'.k_1\ x_1\ (\lambda x_2.k_1'\ x_2\ k_2'))/k\}\ \gamma\ k_2 \\
 & & \text{with } \gamma = \lambda x k_2.k_2\ x
\end{array}
$$

Figure 1: Definitional CPS translation of $\lambda_{\mathcal{S}}$

$$
\begin{array}{rcll}
(\lambda x.t)\ v & = & t\{v/x\} & \beta_{\mathtt{v}} \\
(\lambda x.E[x])\ t & = & E[t] \text{ if } x \notin \mathsf{fv}(E) & \beta_{\Omega} \\
\langle E[\mathcal{S}k.t] \rangle & = & \langle t\{\lambda x.\langle E[x]\rangle/k\}\rangle \text{ if } x \notin \mathsf{fv}(E) & \langle\cdot\rangle_{\mathcal{S}} \\
\langle (\lambda x.t_0)\ \langle t_1 \rangle\rangle & = & (\lambda x.\langle t_0 \rangle)\ \langle t_1 \rangle & \langle\cdot\rangle_{\mathtt{lift}} \\
\langle v \rangle & = & v & \langle\cdot\rangle_{\mathtt{val}} \\
\mathcal{S}k.\langle t \rangle & = & \mathcal{S}k.t & \mathcal{S}_{\langle\cdot\rangle} \\
\lambda x.v\ x & = & v \text{ if } x \notin \mathsf{fv}(v) & \eta_{\mathtt{v}} \\
\mathcal{S}k.k\ t & = & t \text{ if } k \notin \mathsf{fv}(t) & \mathcal{S}_{\mathtt{elim}}
\end{array}
$$

Figure 2: Kameyama and Hasegawa's axiomatization of $\lambda_{\mathcal{S}}$

rest of the computation up to the dynamically nearest enclosing delimiter, and the metacontinuation representing the rest of the computation beyond this delimiter. In the first three equations the metacontinuation k_2 could be η-reduced, yielding Plotkin's familiar CBV CPS translation [69]. In the equation for reset, the current delimited continuation k_1 is moved to the metacontinuation and the delimited term receives the initial delimited continuation. In the equation for shift, the current continuation is captured (and reinitialized) as a lambda abstraction that when applied pushes the then-current delimited continuation on the metacontinuation, and applies the captured continuation to the argument. A CPS-transformed program is run with the initial delimited continuation γ and the identity metacontinuation.

For example, CPS translating the term $x\ \langle y\ \mathcal{S}k.z\ (k\ x')\rangle$ and β-reducing the administrative redexes [19] to avoid clutter, we obtain

$$\lambda k_1 k_2.(\lambda x_2 k_1' k_2'.y\ x_2\ \gamma\ (\lambda x_3.k_1'\ x_3\ k_2'))\ x'\ (\lambda x_1 k_2'.z\ x_1\ \gamma\ k_2')\ (\lambda x_0.x\ x_0\ k_1\ k_2)$$

where the computations are sequentialized according to the evaluation strategy in the source calculus.

The CPS translation for shift and reset induces the following notion of equivalence on $\lambda_{\mathcal{S}}$ terms.

Definition 2.11. Terms t and t' are CPS equivalent, written $t \equiv t'$, if their CPS translations are $\beta\eta$-convertible, where $\beta\eta$-convertibility is the smallest congruence containing the relations $\to_\beta$ and $\to_\eta$:

$$\begin{array}{rcll} (\lambda x.t)\, t' & \to_\beta & t\{t'/x\} & \\ \lambda x.t\, x & \to_\eta & t & \text{if } x \notin \mathsf{fv}(t) \end{array}$$

For example, the reduction rules $t \to_{\mathrm{v}} t'$ given in Section 2.2 are sound w.r.t. CPS because CPS translating t and t' yields $\beta\eta$-convertible terms in the λ-calculus.

The CPS equivalence has been characterized in terms of direct-style equations by Kameyama and Hasegawa, who developed a sound and complete axiomatization of shift and reset [38]: two $\lambda_{\mathcal{S}}$ terms are CPS equivalent iff one can derive their equality using the equations of Figure 2.

The axiomatization is a source of examples for the bisimulation techniques that we study in Sections 4, 5 and 6, and it allows us to relate the notion of CPS equivalence to the notions of contextual equivalence that we introduce in Section 3. In particular, we show that all but one axiom are validated by the bisimilarities for the relaxed semantics, and that all the axioms are validated by the equivalences of the original semantics. The discriminating axiom that confirms the discrepancy between the two semantics is $\mathcal{S}_{\mathtt{elim}}$—the only equation that hinges on the existence of the top-level reset.

It might be possible to consider alternative CPS translations for shift and reset, e.g., as given in [60], that correspond to the relaxed semantics. Such CPS translations require a recursive structure of continuations, which makes it hard to reason about the image of the translations, and, moreover, the operational correspondence between the relaxed semantics and such CPS translations is not as tight as between the original semantics and the original CPS translation considered in this section. Devising a respective axiomatization to be validated by the bisimilarity theories presented in this work is a research path beyond the scope of the present article.

3. Contextual Equivalence

In this section, we discuss the possible definitions of a Morris-style contextual equivalence for the calculus $\lambda_{\mathcal{S}}$. As usual, the idea is to express that two terms are equivalent if and only if they cannot be distinguished when put in an arbitrary context. The question is then what kind of behavior we want to observe. We discuss this issue for the two semantics considered in this paper.

3.1. Definition for the Relaxed Semantics

We first discuss the definition of contextual equivalence for closed terms, before extending it to open terms. As in the regular λ-calculus, we could observe only if a term reduces to a value or not, leading to the following relation.

Definition 3.1. Let t_0, t_1 be closed terms. We write $t_0 \mathbb{C}^1 t_1$ if for all closed C, $C[t_0] \Downarrow_{\mathrm{v}} v_0$ for some v_0 implies $C[t_1] \Downarrow_{\mathrm{v}} v_1$ for some v_1, and conversely for $C[t_1]$.

But in $\lambda_{\mathcal{S}}$, the evaluation of closed terms generates not only values, but also control stuck terms. Taking this into account, a more fine-grained definition of contextual equivalence would be as follows.

Definition 3.2. Let t_0, t_1 be closed terms. We write $t_0 \mathbb{C}^2 t_1$ if for all closed C,

- $C[t_0] \Downarrow_{\mathrm{v}} v_0$ for some v_0 iff $C[t_1] \Downarrow_{\mathrm{v}} v_1$ for some v_1;
- $C[t_0] \Downarrow_{\mathrm{v}} t'_0$ for some control stuck term t'_0 iff $C[t_1] \Downarrow_{\mathrm{v}} t'_1$ for some control stuck term t'_1.

This definition can actually be formulated in a simpler way, where we do not distinguish cases based on the possible normal forms.

Proposition 3.3. *We have $t_0 \mathbb{C}^2 t_1$ iff for all closed C, $C[t_0] \Downarrow_{\mathrm{v}}$ iff $C[t_1] \Downarrow_{\mathrm{v}}$.*

Proof. Suppose that $C[t_0] \Downarrow_{\mathrm{v}}$ iff $C[t_1] \Downarrow_{\mathrm{v}}$ holds. We prove that we have $t_0 \mathbb{C}^2 t_1$ (the reverse implication is immediate). Assume there exists C such that $C[t_0] \Downarrow_{\mathrm{v}} t'_0$ with t'_0 control stuck, and $C[t_1] \Downarrow_{\mathrm{v}} v_1$. Then $C[t_0]\,\Omega \Downarrow_{\mathrm{v}} t'_0\,\Omega$ ($t'_0\,\Omega$ is control stuck), and $C[t_1]\,\Omega \to_{\mathrm{v}}^* v_1\,\Omega \Uparrow_{\mathrm{v}}$. The context $C\,\Omega$ distinguishes t_0 and t_1, hence a contradiction. Therefore, if $C[t_0]$ evaluates to a control stuck term, then so does $C[t_1]$, and similarly for evaluation to values.

By the definitions, it is clear that $\mathbb{C}^2 \subseteq \mathbb{C}^1$. The inclusion is strict, because of terms such as $\mathcal{S}k.\Omega$, which are control stuck terms but diverge when unstuck. Indeed, we have $\mathcal{S}k.\Omega \not\mathbb{C}^2 \Omega$, because $\mathcal{S}k.\Omega$ is a stuck term, but not Ω and, therefore, the second item of Definition 3.2 is violated. However, they are related by $\mathbb{C}^1$.

Proposition 3.4. *We have $\mathcal{S}k.\Omega \mathbb{C}^1 \Omega$.*

Proof. Let C be such that $C[\mathcal{S}k.\Omega] \Downarrow_{\mathrm{v}} v_0$ for some v_0. Then we prove that $C[\Omega]$ reduces to a value as well; in fact, C does not evaluate the term that fills its hole. We define multi-holes contexts H by the following grammar

$$H ::= \Box \mid x \mid \lambda x.H \mid H\,H \mid \mathcal{S}k.H \mid \langle H \rangle$$

and we write $H[t]$ for the plugging of t in all the holes of H. By case analysis on $\to_{\mathrm{v}}$, we can see that if $C[\mathcal{S}k.\Omega] \to_{\mathrm{v}} t'$, then there exists a multi-hole context H such that $t' = H[\mathcal{S}k.\Omega]$ and $C[\Omega] \to_{\mathrm{v}} H[\Omega]$. In particular, we cannot have $C = F[\langle E \rangle]$, otherwise we would have $C[\mathcal{S}k.\Omega] \Uparrow_{\mathrm{v}}$. Consequently, $C[\mathcal{S}k.\Omega] \Downarrow_{\mathrm{v}} v_0$ implies that there exists a mutli-hole context H such that $v_0 = \lambda x.H[\mathcal{S}k.\Omega]$, and $C[\Omega] \Downarrow_{\mathrm{v}} \lambda x.H[\Omega]$. Conversely, if $C[\Omega] \Downarrow_{\mathrm{v}} v_1$, we can prove that $C[\mathcal{S}k.\Omega] \Downarrow_{\mathrm{v}} v$ for some v using the same reasoning. Therefore, we have $\mathcal{S}k.\Omega \mathbb{C}^1 \Omega$.

The relation $\mathbb{C}^2$ is more precise than $\mathbb{C}^1$ about the behavior of terms. Therefore, we chose to work with $\mathbb{C}^2$ as the main contextual equivalence for the relaxed semantics. Henceforth, we simply write $\mathbb{C}$ for $\mathbb{C}^2$.

The relation $\mathbb{C}$ is defined on closed terms, but can be extended to open terms using closing substitutions: we say σ closes t if it maps the free variables of t to closed values. The *open extension* of a relation, written $\mathcal{R}^\circ$, is defined as follows.

Definition 3.5. Let $\mathcal{R}$ be a relation on closed terms, and t_0 and t_1 be open terms. We write $t_0 \mathrel{\mathcal{R}^\circ} t_1$ if for every substitution σ which closes t_0 and t_1, $t_0\sigma \mathrel{\mathcal{R}} t_1\sigma$ holds.

Remark 3.6. Contextual equivalence can be defined directly on open terms by requiring that the context C binds the free variables of the related terms. We prove the resulting relation is equal to $\mathbb{C}^\circ$ in Section 5.4.

To prove completeness of bisimilarities, we use a variant of $\mathbb{C}$ which takes into account only evaluation contexts to compare terms.

Definition 3.7. Let t_0, t_1 be closed terms. We write $t_0 \mathrel{\mathbb{D}} t_1$ if for all closed F,

- $F[t_0] \Downarrow_{\mathrm{v}} v_0$ for some v_0 iff $F[t_1] \Downarrow_{\mathrm{v}} v_1$ for some v_1;
- $F[t_0] \Downarrow_{\mathrm{v}} t'_0$ for some control stuck term t'_0 iff $F[t_1] \Downarrow_{\mathrm{v}} t'_1$ for some control stuck term t'_1.

The definitions imply $\mathbb{C} \subseteq \mathbb{D}$. While proving completeness of applicative bisimilarity in Section 5, we also prove $\mathbb{D} = \mathbb{C}$, which means that testing with evaluation contexts is as discriminating as testing with any contexts. Such a simplification result is similar to Milner's context lemma [62].

The relations $\mathbb{C}^1$ and $\mathbb{C}^2$ are not suitable for the original semantics, because they distinguish terms that should be equated according to Kameyama and Hasegawa's axiomatization. Indeed, according to these relations, $\mathcal{S}k.k\, v$ (where $k \notin \mathsf{fv}(v)$) cannot be related to v (axiom $\mathcal{S}_{\mathtt{elim}}$ in Figure 2), because a stuck term cannot be related to a value. In the next section, we discuss a definition of contextual equivalence for the original semantics.

3.2. Definition for the Original Semantics

To reflect the fact that in the original semantics terms are evaluated within an enclosing reset, the contextual equivalence we consider for the original semantics tests terms in contexts of the form $\langle C \rangle$ only. Because delimited terms cannot reduce to stuck terms (Proposition 2.9), the only possible observable action is evaluation to values. We, therefore, define contextual equivalence for the original semantics as follows.

Definition 3.8. Let t_0, t_1 be closed terms. We write $t_0 \mathrel{\mathbb{P}} t_1$ if for all closed C, $\langle C[t_0] \rangle \Downarrow_{\mathrm{v}} v_0$ for some v_0 iff $\langle C[t_1] \rangle \Downarrow_{\mathrm{v}} v_1$ for some v_1.

The relation $\mathbb{P}$ is defined on all (closed) terms, not just delimited ones. The resulting relation is less discriminating than $\mathbb{C}$, because $\mathbb{P}$ uses contexts of a particular form, while $\mathbb{C}$ tests with all contexts.

Proposition 3.9. *We have* $\mathbb{C} \subseteq \mathbb{P}$*.*

As a result, any equivalence between terms we prove for the relaxed semantics also holds in the original semantics, and any bisimilarity sound w.r.t. $\mathbb{C}$ (like the bisimilarities we define in Sections 4, 5, and 6.1) is also sound w.r.t. $\mathbb{P}$. However, to reach completeness, we have to design a bisimilarity suitable for delimited terms (see Section 6.5). As for the relaxed semantics, we extend $\mathbb{P}$ to open terms using Definition 3.5.

The inclusion of Proposition 3.9 is strict; in particular, $\mathbb{P}$ verifies the axiom $\mathcal{S}_{\texttt{elim}}$, while $\mathbb{C}$ does not. In fact, we prove in Section 4.7 that $\mathbb{P}$ contains the CPS equivalence $\equiv$. The reverse inclusion holds neither for $\mathbb{P}$ nor $\mathbb{C}$: there exist contextually equivalent terms that are not CPS equivalent.

Proposition 3.10.

1. *We have* $\Omega \mathrel{\mathbb{P}} \Omega\Omega$ *(respectively* $\Omega \mathrel{\mathbb{C}} \Omega\Omega$*), but* $\Omega \not\equiv \Omega\Omega$*.*
2. *We have* $\Theta \mathrel{\mathbb{P}} \Delta$ *(respectively* $\Theta \mathrel{\mathbb{C}} \Delta$*), but* $\Theta \not\equiv \Delta$*.*

The contextual equivalences $\mathbb{C}$ and $\mathbb{P}$ put all diverging terms in one equivalence class, while CPS equivalence is more discriminating. Furthermore, as is usual with equational theories for λ-calculi, CPS equivalence is not strong enough to equate Turing's and Curry's (call-by-value) fixed-point combinators.

As explained in the introduction, contextual equivalence is difficult to prove in practice for two given terms because of the quantification over contexts. We look for a suitable replacement (that is, an equivalence that is at least sound w.r.t. $\mathbb{C}$ or $\mathbb{P}$) by studying different styles of bisimulation in the next sections.

4. Normal-Form Bisimilarity

Normal-form bisimilarity [52] (originally defined in [72], where it was called *open bisimilarity*) equates (open) terms by reducing them to normal form, and then requiring the sub-terms of these normal forms to be bisimilar. Unlike applicative and environmental bisimilarities (studied in the next sections), normal-form bisimilarity usually does not contain a universal quantification over testing terms or contexts in its definition, and is therefore easier to use than the former two. However, it is also usually not complete w.r.t. contextual equivalence, meaning that there exist contextually equivalent terms that are not normal-form bisimilar.

A notion of normal-form bisimulation has been defined in various calculi, including the pure λ-calculus [51, 52], the λ-calculus with ambiguous choice [53], the $\lambda\mu$-calculus [54], and the $\lambda\mu\rho$-calculus [80], a calculus with control and store, where normal-form bisimilarity characterizes contextual equivalence. It has also been defined for typed languages [55, 56]. In this section, we discuss how we can define normal-form bisimilarity for the relaxed semantics of $\lambda_{\mathcal{S}}$, and then propose up-to techniques and other improvements. We then define a normal-form bisimilarity dedicated to the original semantics.

Definition of $\star$ on values:

$$x \star y \stackrel{\mathsf{def}}{=} x\, y \qquad \lambda x.t \star y \stackrel{\mathsf{def}}{=} t\{y/x\}$$

Definitions of $\mathcal{R}^{\mathsf{NF}\eta}$ on normal forms and of $\mathcal{R}^{\mathsf{C}}$ on contexts

$$\frac{E_0[x]\ \mathcal{R}\ E_1[x] \qquad x \text{ fresh}}{E_0\ \mathcal{R}^{\mathsf{C}}\ E_1} \qquad \frac{\langle E_0[x]\rangle\ \mathcal{R}\ \langle E_1[x]\rangle \qquad F_0[x]\ \mathcal{R}\ F_1[x] \qquad x \text{ fresh}}{F_0[\langle E_0\rangle]\ \mathcal{R}^{\mathsf{C}}\ F_1[\langle E_1\rangle]}$$

$$\frac{v_0 \star x\ \mathcal{R}\ v_1 \star x \qquad x \text{ fresh}}{v_0\ \mathcal{R}^{\mathsf{NF}\eta}\ v_1} \qquad \frac{E_0\ \mathcal{R}^{\mathsf{C}}\ E_1 \qquad \langle t_0\rangle\ \mathcal{R}\ \langle t_1\rangle}{E_0[\mathcal{S}k.t_0]\ \mathcal{R}^{\mathsf{NF}\eta}\ E_1[\mathcal{S}k.t_1]}$$

$$\frac{F_0\ \mathcal{R}^{\mathsf{C}}\ F_1 \qquad v_0\ \mathcal{R}^{\mathsf{NF}\eta}\ v_1}{F_0[x\, v_0]\ \mathcal{R}^{\mathsf{NF}\eta}\ F_1[x\, v_1]}$$

Figure 3: Definitions of the operator $\star$ and of the relations $\mathcal{R}^{\mathsf{NF}\eta}$ and $\mathcal{R}^{\mathsf{C}}$

4.1. Definition

In the λ-calculus [72, 52], the definition of normal-form bisimilarity has to take into account only values and open stuck terms. In $\lambda_{\mathcal{S}}$ with the relaxed semantics, we have to relate also control stuck terms; we propose here a first way to deal with these terms, that will be refined in the next subsection. Deconstructing normal forms leads to comparing contexts as well as terms. Given a relation $\mathcal{R}$ on terms, we define in Figure 3 an extension of $\mathcal{R}$ to normal forms, written $\mathcal{R}^{\mathsf{NF}\eta}$, which relies on an application operator for values $\star$ and on a relation $\mathcal{R}^{\mathsf{C}}$ on contexts. The rationale behind these definitions becomes clear when we explain our notion of normal-form bisimilarity, defined below.

Definition 4.1. A relation $\mathcal{R}$ on terms is a normal-form simulation if $t_0\ \mathcal{R}\ t_1$ and $t_0 \Downarrow_{\mathrm{v}} t'_0$ implies that there exists t'_1 such that $t_1 \Downarrow_{\mathrm{v}} t'_1$ and $t'_0\ \mathcal{R}^{\mathsf{NF}\eta}\ t'_1$. A relation $\mathcal{R}$ is a normal-form bisimulation if both $\mathcal{R}$ and $\mathcal{R}^{-1}$ are normal-form simulations. Normal-form bisimilarity, written $\mathbb{N}$, is the largest normal-form bisimulation.

In this section, we often drop the "normal-form" attribute when it does not cause confusion. Two terms t_0 and t_1 are bisimilar if their evaluations lead to matching normal forms (e.g., if t_0 evaluates to a control stuck term, then so does t_1) with bisimilar sub-components. We now detail the different cases.

Normal-form bisimilarity does not distinguish between evaluation to a variable and evaluation to a λ-abstraction. Instead, we relate terms evaluating to any values v_0 and v_1 by comparing $v_0 \star x$ and $v_1 \star x$, where x is fresh. As originally pointed out by Lassen [52], this is necessary for the bisimilarity to be sound w.r.t. η-expansion; otherwise it would distinguish η-equivalent terms such as $\lambda y.x\, y$ and x. Using $\star$ instead of regular application avoids the introduction of unnecessary β-redexes, which could reveal themselves problematic in proofs.

For a control stuck term $E_0[\mathcal{S}k.t_0]$ to be executed, it has to be plugged into a pure evaluation context surrounded by a reset; by doing so, we obtain a term of the form $\langle t_0\{\lambda x.\langle E_0'[x]\rangle/k\}\rangle$ for some context E_0'. Notice that the resulting term is within a reset; similarly, when comparing $E_0[\mathcal{S}k.t_0]$ and $E_1[\mathcal{S}k.t_1]$, we ask for the shift bodies t_0 and t_1 to be related when surrounded by a reset. We also compare E_0 and E_1, which amounts to executing $E_0[x]$ and $E_1[x]$ for a fresh x, since the two contexts are pure. Comparing t_0' and t_1' without reset would be too discriminating, as it would distinguish contextually equivalent terms such as $\mathcal{S}k.\langle t\rangle$ and $\mathcal{S}k.t$ (axiom $\mathcal{S}_{\langle\cdot\rangle}$). Indeed, without reset, we would have to relate $\langle t\rangle$ and t, which are not equivalent in general (take $t = \mathcal{S}k'.v$ for some v), while Definition 4.1 requires $\langle\langle t\rangle\rangle$ and $\langle t\rangle$ to be related (which holds for all t; see Example 4.3).

The open stuck terms $F_0[x\, v_0]$ and $F_1[x\, v_1]$ are bisimilar if the values v_0 and v_1 as well as the contexts F_0 and F_1 are related. We have to be careful when defining bisimilarity on (possibly non pure) evaluation contexts. We cannot simply relate F_0 and F_1 by executing $F_0[y]$ and $F_1[y]$ for a fresh y. Such a definition would equate the contexts $\square$ and $\langle\square\rangle$, which in turn would relate the terms $x\, v$ and $\langle x\, v\rangle$, which are not contextually equivalent: they are distinguished by the context $(\lambda x.\square)\, \lambda y.\mathcal{S}k.\Omega$. A context containing a reset enclosing the hole should be related only to contexts with the same property. However, we do not want to precisely count the number of delimiters around the hole; doing so would distinguish $\langle\square\rangle$ and $\langle\langle\square\rangle\rangle$, and, therefore, it would discriminate the contextually equivalent terms $\langle x\, v\rangle$ and $\langle\langle x\, v\rangle\rangle$. Hence, the definition of $\mathcal{R}^{\mathsf{C}}$ (Figure 3) checks that if one of the contexts contains a reset surrounding the hole, then so does the other; then it compares the contexts beyond the first enclosing delimiter by simply evaluating them using a fresh variable. As a result, it rightfully distinguishes $\square$ and $\langle\square\rangle$, but it relates $\langle\square\rangle$ and $\langle\langle\square\rangle\rangle$.

As a first basic result about normal-form bisimilarity, we show that $\rightarrow_{\mathrm{v}}$ (and hence, $\Downarrow_{\mathrm{v}}$) is included in $\mathbb{N}$.

Proposition 4.2. *If $t \rightarrow_{\mathrm{v}} t'$, then $t \,\mathbb{N}\, t'$.*

Proof. Because the calculus is deterministic, $t \Downarrow_{\mathrm{v}} t''$ iff $t' \Downarrow_{\mathrm{v}} t''$, and it is easy to check that the identity relation $\{(t,t) \mid t \in \mathcal{T}\}$ is a normal-form bisimulation.

We now give some examples to show how to prove equivalences using normal-form bisimulation.

Example 4.3 (double reset). We prove that $\langle t\rangle \,\mathbb{N}\, \langle\langle t\rangle\rangle$ by showing that $\mathcal{R} \stackrel{\mathsf{def}}{=} \{(\langle t\rangle, \langle\langle t\rangle\rangle) \mid t \in \mathcal{T}\} \cup \mathbb{N}$ is a bisimulation. First, note that the case $\langle t\rangle \Downarrow_{\mathrm{v}} E[\mathcal{S}k.t']$ is not possible because of Proposition 2.9. Then, we prove that $\langle t\rangle \Downarrow_{\mathrm{v}} v$ iff $\langle\langle t\rangle\rangle \Downarrow_{\mathrm{v}} v$. If $\langle t\rangle \Downarrow_{\mathrm{v}} v$, then $\langle\langle t\rangle\rangle \rightarrow_{\mathrm{v}}^{*} \langle v\rangle \rightarrow_{\mathrm{v}} v$. Conversely, if $\langle\langle t\rangle\rangle \Downarrow_{\mathrm{v}} v$, then $\langle t\rangle$ cannot diverge or cannot reduce to an open stuck term (otherwise, $\langle\langle t\rangle\rangle$ would also diverge or reduce to an open stuck term). Hence, we have $\langle t\rangle \Downarrow_{\mathrm{v}} v'$, which entails $\langle\langle t\rangle\rangle \rightarrow_{\mathrm{v}}^{*} \langle v'\rangle \rightarrow_{\mathrm{v}} v'$, which in turn implies $v = v'$ because normal forms are unique. Consequently, we have $\langle t\rangle \Downarrow_{\mathrm{v}} v$ iff $\langle\langle t\rangle\rangle \Downarrow_{\mathrm{v}} v$, and $v \ \mathbb{N}^{\mathsf{NF}\eta}\ v$ holds.

If $\langle t\rangle \Downarrow_{\mathrm{v}} F[x\,v]$, then by Proposition 2.9, there exists F' such that $F = \langle F'\rangle$. Therefore, we have $\langle\langle t\rangle\rangle \Downarrow_{\mathrm{v}} \langle\langle F'[x\,v]\rangle\rangle$. We have $v\ \mathbb{N}^{\mathsf{NF}\eta}\ v$, and we have to prove that $\langle F'\rangle\ \mathcal{R}^{\mathsf{C}}\ \langle\langle F'\rangle\rangle$ holds to conclude. If F' is a pure context E, then we have to prove $\langle E[y]\rangle\ \mathcal{R}\ \langle E[y]\rangle$ and $y\ \mathcal{R}\ \langle y\rangle$ for a fresh y, which are both true because $\mathbb{N} \subseteq \mathcal{R}$. If $F' = F''[\langle E\rangle]$, then given a fresh y, we have to prove $\langle F''[y]\rangle\ \mathcal{R}\ \langle\langle F''[y]\rangle\rangle$ (clear by the definition of $\mathcal{R}$), and $\langle E[y]\rangle\ \mathcal{R}\ \langle E[y]\rangle$ (true because $\mathbb{N} \subseteq \mathcal{R}$).

Similarly, if $\langle\langle t\rangle\rangle \Downarrow_{\mathrm{v}} F[x\,v]$, then by Proposition 2.9, there exists F' such that $F = \langle F'\rangle$. Then $\langle t\rangle$ cannot evaluate to a control stuck term (because it is a delimited term), and it cannot evaluate to a value (otherwise, $\langle\langle t\rangle\rangle$ would evaluate to this value). Therefore, we have $\langle t\rangle \Downarrow_{\mathrm{v}} F'[x\,v]$. In turn, this implies that there exists F'' such that $F' = \langle F''\rangle$ (using Proposition 2.9 again). Consequently, we have $\langle\langle t\rangle\rangle \Downarrow_{\mathrm{v}} \langle\langle F''[x\,v]\rangle\rangle$ and $\langle t\rangle \Downarrow_{\mathrm{v}} \langle F''[x\,v]\rangle$, so we can conclude as in the previous case.

The relation $\mathcal{R}$ is therefore a bisimulation, meaning that $\mathcal{R} \subseteq \mathbb{N}$. Because we have $\langle t\rangle\ \mathcal{R}\ \langle\langle t\rangle\rangle$, we also have $\langle t\rangle\ \mathbb{N}\ \langle\langle t\rangle\rangle$, as wished.

Example 4.4 (fixed-point combinators). We study here the relationships between Turing's and Curry's fixed-point combinator and their respective variants with delimited control [17] (see Example 2.2 for the definitions). First, we prove that Turing's combinator Θ is bisimilar to its variant $\Theta_{\mathcal{S}} \stackrel{\mathsf{def}}{=} \langle \theta\ \mathcal{S}k.k\ k\rangle$. We build the candidate relation $\mathcal{R}$ incrementally, starting from $(\Theta, \Theta_{\mathcal{S}})$. Evaluating these two terms, we obtain

$$\Theta \Downarrow_{\mathrm{v}} \lambda y.y\ (\lambda z.\theta\ \theta\ y\ z) \stackrel{\mathsf{def}}{=} v_0, \text{ and}$$

$$\Theta_{\mathcal{S}} \Downarrow_{\mathrm{v}} \lambda y.y\ (\lambda z.(\lambda x.\langle \theta\ x\rangle)\ (\lambda x.\langle \theta\ x\rangle)\ y\ z) \stackrel{\mathsf{def}}{=} v_1.$$

We, therefore, extend $\mathcal{R}$ with $(v_0 \star y, v_1 \star y)$, where y is fresh. These two new terms are open stuck, so we add their decomposition to $\mathcal{R}$. Let $v_0' \stackrel{\mathsf{def}}{=} \lambda z.\theta\ \theta\ y\ z$ and $v_1' \stackrel{\mathsf{def}}{=} \lambda z.(\lambda x.\langle \theta\ x\rangle)\ (\lambda x.\langle \theta\ x\rangle)\ y\ z$; then we add $(v_0' \star z, v_1' \star z)$ and (z, z) for a fresh z to $\mathcal{R}$. Evaluating $v_0' \star z$ and $v_1' \star z$, we obtain respectively $y\ v_0'\ z$ and $y\ v_1'\ z$; to relate these two open stuck terms, we just need to add $(x\ z, x\ z)$ (for a fresh x) to $\mathcal{R}$, since we already have $v_0'\ \mathcal{R}^{\mathsf{NF}\eta}\ v_1'$. The constructed relation $\mathcal{R}$ we obtain is a normal-form bisimulation.

In contrast, Curry's combinator Δ is not bisimilar to its delimited-control variant $\Delta_{\mathcal{S}} \stackrel{\mathsf{def}}{=} \lambda x.\langle \delta_x\ \mathcal{S}k.k\ k\rangle$. Indeed, evaluating the bodies of the two values, we obtain respectively $x\,(\lambda z.\delta_x\,\delta_x\,z)$ and $\langle\langle x\ (\lambda z.(\lambda y.\langle \delta_x\ y\rangle)\ (\lambda y.\langle \delta_x\ y\rangle)\ z)\rangle\rangle$, and these open stuck terms are not bisimilar, because $\square$ is not related to $\langle\langle \square\rangle\rangle$ by $\mathbb{N}^{\mathsf{C}}$. In fact, Δ and $\Delta_{\mathcal{S}}$ are not contextually equivalent, as they are distinguished by the context $\square\ \lambda x.\mathcal{S}k.\Omega$. Finally, we can prove that the two original combinators Θ and Δ are bisimilar, using the same bisimulation as in [52].

4.2. Soundness and Completeness

Usual congruence proofs for normal-form bisimilarities include direct proofs, where a context and/or substitutive closure of the bisimilarity is proved to be itself a bisimulation [51, 53, 80], or proofs based on CPS translations [52, 54]. The CPS approach consists in proving a CPS-based correspondence between the bisimilarity $\mathcal{R}_1$ we want to prove sound and a relation $\mathcal{R}_2$ that we already know is a congruence. Because CPS translations are usually themselves compatible, we can then conclude that $\mathcal{R}_1$ is a congruence. For example, for the λ-calculus, Lassen proved a CPS-correspondence between eager normal-form bisimilarity and Böhm trees equivalence [52]. In this section, we use the relaxed semantics, but the CPS translation corresponds to the original one; we, therefore, cannot rely on a CPS-based proof method.

Instead, we follow a direct approach and define a closure that we prove to be a bisimulation. We only sketch the proof here, because the complete proof (available in Appendix A.1) requires a lot of auxiliary notations and definitions. The idea is to consider pairs of terms of the form $(\overrightarrow{F_0}[t_0]\overrightarrow{\sigma_0}, \overrightarrow{F_1}[t_1]\overrightarrow{\sigma_1})$, where $\overrightarrow{F_0}$, $\overrightarrow{F_1}$, (respectively t_0, t_1, and $\overrightarrow{\sigma_0}$, $\overrightarrow{\sigma_1}$) are contexts (respectively terms, and substitutions) pairwise related by $\mathbb{N}$ (with some extra conditions), and show that they are normal-form bisimilar. This implies that $\mathbb{N}$ is substitutive and compatible w.r.t. evaluation contexts. We can then prove separately compatibility w.r.t. λ-abstraction and shift easily.

Theorem 4.5. *The relation* $\mathbb{N}$ *is a congruence.*

We deduce that $\mathbb{N}$ is sound w.r.t. contextual equivalence.

Theorem 4.6. *We have* $\mathbb{N} \subseteq \mathbb{C}$.

The following counter-example shows that the inclusion is in fact strict; normal-form bisimilarity is not complete.

Proposition 4.7. *Let* $i \stackrel{\mathsf{def}}{=} \lambda y.y$. *We have* $\langle\langle x\, i\rangle\, \mathcal{S}k.i\rangle \;\mathbb{C}^\circ\; \langle\langle x\, i\rangle\, (\langle x\, i\rangle\, \mathcal{S}k.i)\rangle$, *but* $\langle\langle x\, i\rangle\, \mathcal{S}k.i\rangle \not\mathbb{N} \langle\langle x\, i\rangle\, (\langle x\, i\rangle\, \mathcal{S}k.i)\rangle$.

Proof. We prove that $\langle\langle x\, i\rangle\, \mathcal{S}k.i\rangle \;\mathbb{C}^\circ\; \langle\langle x\, i\rangle\, (\langle x\, i\rangle\, \mathcal{S}k.i)\rangle$ holds using applicative bisimilarity in Proposition 5.19. They are not normal-form bisimilar, because the terms $\langle y\, \mathcal{S}k.i\rangle$ and $\langle y\, (\langle x\, i\rangle\, \mathcal{S}k.i)\rangle$ (where y is fresh) are not bisimilar: the former evaluates to i while the latter is in normal form (but is not a value).

Remark 4.8. Following Filinski's simulation of shift and reset in terms of $\mathsf{call/cc}$ and a single reference cell [26], one can express the terms of the $\lambda_{\mathcal{S}}$-calculus in the $\lambda\mu\rho$-calculus [80]. Yet, Støvring and Lassen's normal-form bisimilarity is sound and complete with respect to the contextual equivalence of $\lambda\mu\rho$ [80] (a calculus with store and a construct similar to $\mathsf{call/cc}$), while our relation is only sound. It shows that $\lambda\mu\rho$ is more expressive and can distinguish more terms than $\lambda_{\mathcal{S}}$. For example, the encodings of the two terms of Proposition 4.7 in $\lambda\mu\rho$ would not be contextually equivalent in $\lambda\mu\rho$, since substituting for x a value that, e.g., increments a value of some reference cell, would lead to two different states that can be easily distinguished observationally. A more general and precise characterization of the relation between the two calculi is an interesting question, but it falls out of the scope of the present article.

4.3. Up-to Techniques

The idea behind up-to techniques [75, 49, 73] is to define relations that are not exactly bisimulations but are included in bisimulations. It usually leads to definitions of simpler candidate relations and to simpler bisimulation proofs. Here, we discuss only bisimulation up to context (with also some limited form of up to reduction), one of the most powerful up-to technique, which allows to abstract away a common context during a bisimulation proof.

In contrast with normal-form bisimulation, the notion of bisimulation up to context we introduce does not respect η-expansion; we discuss why in Remark 4.14. To simplify the soundness proof, we also introduce some limited use of up to reduction in the definition. Given a relation $\mathcal{R}$ on terms, we write $t_0 \mathrel{\mathcal{R}^{\searrow}} t_1$ if there exist t_0', t_1' such that $t_0 \to_{\mathrm{v}}^* t_0'$, $t_1 \to_{\mathrm{v}}^* t_1'$, and $t_0' \mathrel{\mathcal{R}} t_1'$. We also define relations $\mathcal{R}^{\mathsf{NF}\searrow}$ and normal forms and $\mathcal{R}^{\mathsf{C}\searrow}$ on contexts as follows:

$$\frac{E_0[x] \mathrel{\mathcal{R}} E_1[x] \qquad x \text{ fresh}}{E_0 \mathrel{\mathcal{R}^{\mathsf{C}\searrow}} E_1} \qquad \frac{\langle E_0[x]\rangle \mathrel{\mathcal{R}} \langle E_1[x]\rangle \qquad F_0[x] \mathrel{\mathcal{R}^{\searrow}} F_0[x] \qquad x \text{ fresh}}{F_0[\langle E_0\rangle] \mathrel{\mathcal{R}^{\mathsf{C}\searrow}} F_1[\langle E_1\rangle]}$$

$$\frac{E_0 \mathrel{\mathcal{R}^{\mathsf{C}\searrow}} E_1 \qquad \langle t_0\rangle \mathrel{\mathcal{R}} \langle t_1\rangle}{E_0[\mathcal{S}k.t_0] \mathrel{\mathcal{R}^{\mathsf{NF}\searrow}} E_1[\mathcal{S}k.t_1]} \qquad \frac{F_0 \mathrel{\mathcal{R}^{\mathsf{C}\searrow}} F_1 \qquad v_0 \mathrel{\mathcal{R}^{\mathsf{NF}\searrow}} v_1}{F_0[x\, v_0] \mathrel{\mathcal{R}^{\mathsf{NF}\searrow}} F_1[x\, v_1]}$$

$$\frac{}{x \mathrel{\mathcal{R}^{\mathsf{NF}\searrow}} x} \qquad \frac{t_0 \mathrel{\mathcal{R}} t_1}{\lambda x.t_0 \mathrel{\mathcal{R}^{\mathsf{NF}\searrow}} \lambda x.t_1}$$

The definitions of $\mathcal{R}^{\mathsf{C}\searrow}$ and $\mathcal{R}^{\mathsf{NF}\searrow}$ are the same as $\mathcal{R}^{\mathsf{C}}$ and $\mathcal{R}^{\mathsf{NF}\eta}$ on pure contexts, and on control and open stuck terms, respectively. Values are related without using $\star$, which means that two η-equivalent values can no longer be related. Finally, the clause for general evaluation contexts uses up to reduction in one of the premises.

Next, we define the substitutive, reflexive, and context closure $\widehat{\mathcal{R}}$ of a relation $\mathcal{R}$ by the rules of Figure 4. Note that we have two rules for compatibility w.r.t. evaluation contexts: two terms that are not both delimited can be plugged only in contexts F_0, F_1 related by $\mathcal{R}^{\mathsf{C}\searrow}$. Two delimited terms may in addition be put into contexts F_0, F_1 verifying $F_0[x] \mathrel{\mathcal{R}^{\searrow}} F_0[x]$ for a fresh x, which is a weaker constraint than $F_0 \mathrel{\mathcal{R}^{\mathsf{C}\searrow}} F_1$. This extra case for delimited terms helps in the proof of soundness of the technique, and we remind it would not be sound to put any terms in contexts verifying only $F_0[x] \mathrel{\mathcal{R}^{\searrow}} F_0[x]$, as explained in Section 4.1.

We can now define bisimulation up to substitutive, reflexive, and context closure (in short, up to context) as follows.

$$\frac{}{t \mathrel{\widehat{\mathcal{R}}} t} \qquad \frac{t_0 \mathrel{\mathcal{R}} t_1}{t_0 \mathrel{\widehat{\mathcal{R}}} t_1} \qquad \frac{t_0 \mathrel{\widehat{\mathcal{R}}} t_1 \qquad v_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1}{t_0\{v_0/x\} \mathrel{\widehat{\mathcal{R}}} t_1\{v_1/x\}} \qquad \frac{t_0 \mathrel{\widehat{\mathcal{R}}} t_1 \qquad F_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{C}\searrow}} F_1}{F_0[t_0] \mathrel{\widehat{\mathcal{R}}} F_1[t_1]}$$

$$\frac{t_0 \mathrel{\widehat{\mathcal{R}}} t_1 \qquad t_0, t_1 \text{ delimited} \qquad F_0[x] \mathrel{\widehat{\mathcal{R}}^{\searrow}} F_1[x] \qquad x \text{ fresh}}{F_0[t_0] \mathrel{\widehat{\mathcal{R}}} F_1[t_1]}$$

$$\frac{t_0 \mathrel{\widehat{\mathcal{R}}} t_1}{\lambda x.t_0 \mathrel{\widehat{\mathcal{R}}} \lambda x.t_1} \qquad \frac{t_0 \mathrel{\widehat{\mathcal{R}}} t_1}{\mathcal{S}k.t_0 \mathrel{\widehat{\mathcal{R}}} \mathcal{S}k.t_1}$$

Figure 4: Substitutive, reflexive, and context closure of a relation $\mathcal{R}$

Definition 4.9. A relation $\mathcal{R}$ on terms is a normal-form simulation up to context if $t_0 \mathrel{\mathcal{R}} t_1$ and $t_0 \Downarrow_{\mathrm{v}} t_0'$ implies that there exists t_1' such that $t_1 \Downarrow_{\mathrm{v}} t_1'$ and $t_0' \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}}} t_1'$. A relation $\mathcal{R}$ is a normal-form bisimulation up to context if both $\mathcal{R}$ and $\mathcal{R}^{-1}$ are normal-form simulations up to context.

In a bisimulation up to context $\mathcal{R}$, the results of the evaluations of two terms are compared using the closure $\widehat{\mathcal{R}}$, instead of using simply $\mathcal{R}$ itself. Because $\widehat{\mathcal{R}}$ is larger than $\mathcal{R}$, it is easier to obtain terms related by $\widehat{\mathcal{R}}$ rather than $\mathcal{R}$. For example, we can simplify the proof of bisimilarity between Turing's fixed point combinator Θ and its delimited-control variant $\Theta_{\mathcal{S}}$ (cf. Example 4.4).

Example 4.10 (fixed-point combinators). The relation

$$\mathcal{R} \stackrel{\mathsf{def}}{=} \{(\Theta, \Theta_{\mathcal{S}}), (\Theta, (\lambda x.\langle \theta\, x\rangle)\, (\lambda x.\langle \theta\, x\rangle))\}$$

is a bisimulation up to context. Indeed, we remind that

$$\Theta \Downarrow_{\mathrm{v}} \lambda y.y\, (\lambda z.\Theta\, y\, z) \stackrel{\mathsf{def}}{=} v_0, \text{ and}$$
$$\Theta_{\mathcal{S}} \Downarrow_{\mathrm{v}} \lambda y.y\, (\lambda z.(\lambda x.\langle \theta\, x\rangle)\, (\lambda x.\langle \theta\, x\rangle)\, y\, z) \stackrel{\mathsf{def}}{=} v_1.$$

The bodies of v_0 and v_1 share the common context $y\, (\lambda z.\Box\, y\, z)$, and the two terms filling the holes (respectively Θ and $(\lambda x.\langle \theta\, x\rangle)\, (\lambda x.\langle \theta\, x\rangle)$) are in $\mathcal{R}$, so we have $v_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}}} v_1$. The terms Θ and $(\lambda x.\langle \theta\, x\rangle)\, (\lambda x.\langle \theta\, x\rangle)$ also reduce respectively to v_0 and v_1, hence the result holds.

We now prove the soundness of the up to context technique. For this, we define *non-η bisimulation* up to reduction, an up-to relation which uses $\mathcal{R}^{\mathsf{NF}\searrow}$ instead of $\mathcal{R}^{\mathsf{NF}\eta}$.

Definition 4.11. A relation $\mathcal{R}$ on terms is a non-η simulation up to reduction if $t_0 \mathrel{\mathcal{R}} t_1$ and $t_0 \Downarrow_{\mathrm{v}} t_0'$ implies that there exists t_1' such that $t_1 \Downarrow_{\mathrm{v}} t_1'$ and $t_0' \mathrel{\mathcal{R}^{\mathsf{NF}\searrow}} t_1'$. A relation $\mathcal{R}$ is a non-η bisimulation up to reduction if both $\mathcal{R}$ and $\mathcal{R}^{-1}$ are non-η simulations up to reduction.

Lemma 4.12. *If $\mathcal{R}$ is a non-η bisimulation up to reduction, then $\mathcal{R} \subseteq \mathbb{N}$.*

PROOF. By showing that $\{(t_0, t_1) \mid \exists t'_0, t'_1, t_0 \rightarrow^*_{\mathrm{v}} t'_0 \wedge t_1 \rightarrow^*_{\mathrm{v}} t'_1 \wedge t'_0 \mathrel{\mathcal{R}} t'_1\}$ is a normal-form bisimulation.

Proposition 4.13. *If $\mathcal{R}$ is a normal-form bisimulation up to context, then $\widehat{\mathcal{R}}$ is a non-η bisimulation.*

More precisely, we prove in Appendix A.2 that if $t_0 \mathrel{\widehat{\mathcal{R}}} t_1$ and $t_0 \Downarrow_{\mathrm{v}} t'_0$ in m steps or less, then there exists t'_1 such that $t_1 \Downarrow_{\mathrm{v}} t'_1$ and $t'_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} t'_1$. The proof is by induction on m and the definition of $t_0 \mathrel{\widehat{\mathcal{R}}} t_1$, ordered lexicographically. As a result, if $\mathcal{R}$ is a bisimulation up to context, and if $t_0 \mathrel{\mathcal{R}} t_1$, then $t_0 \mathrel{\mathbb{N}} t_1$, because $\mathcal{R} \subseteq \widehat{\mathcal{R}} \subseteq \mathbb{N}$.

Remark 4.14 (η-expansion). We cannot prove Proposition 4.13 if we use $\star$ in $\mathcal{R}^{\mathsf{NF}\searrow}$, as in $\mathcal{R}^{\mathsf{NF}\eta}$. With η-expansion, one of the inductive cases of the proof would require a result stating that $x\,y \mathrel{\widehat{\mathcal{R}}} t$ implies $t \Downarrow_{\mathrm{v}} t'$ and $x\,y \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\eta}} t'$. But proving this result is as difficult as proving Proposition 4.13 directly. This issue is not specific to the calculus, as it arises in the plain λ-calculus as well, but to the proof technique. The only proof of soundness of an up to context technique which respects η-expansion can be found in [52], where the author uses a CPS translation, not a direct proof (see our discussion about CPS translations vs direct proofs in Section 4.2).

When proving equivalence of terms, it is sometimes easier to reason in a small-step fashion instead of trying to evaluate terms completely. To allow this kind of reasoning, we define small-step bisimulation up to context as follows.

Definition 4.15. A relation $\mathcal{R}$ on terms is a small-step normal-form simulation up to context if $t_0 \mathrel{\mathcal{R}} t_1$ implies:

- if $t_0 \rightarrow_{\mathrm{v}} t'_0$, then there exists t'_1 such that $t_1 \rightarrow^*_{\mathrm{v}} t'_1$ and $t'_0 \mathrel{\widehat{\mathcal{R}}^{\searrow}} t'_1$;
- if t_0 is a normal form, then there exists t'_1 such that $t_1 \Downarrow_{\mathrm{v}} t'_1$ and $t_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} t'_1$.

A relation $\mathcal{R}$ is a small-step normal-form bisimulation up to context if both $\mathcal{R}$ and $\mathcal{R}^{-1}$ are small-step normal-form simulations up to context.

Note that the small step style is not specific to up to context, and can be used also with Definition 4.1. We can easily adapt the proof of Proposition 4.13 to prove the soundness of small step bisimulation up to context.

Proposition 4.16. *If $\mathcal{R}$ is a small-step normal-form bisimulation up to context, then $\widehat{\mathcal{R}}$ is a non-η bisimulation up to reduction.*

The next example demonstrates how useful small-step relations can be.

Example 4.17. We prove that if $x \notin \mathsf{fv}(E)$, then $\langle (\lambda x.\langle E[x]\rangle)\, t\rangle \mathrel{\mathbb{N}} \langle E[t]\rangle$. Let $\mathcal{R} \stackrel{\mathsf{def}}{=} \{(\langle(\lambda x.\langle E[x]\rangle)\, t\rangle, \langle E[t]\rangle) \mid t \in \mathcal{T}, E \in \mathcal{PC}, x \notin \mathsf{fv}(E)\}$. We prove that $\mathcal{R} \cup \mathbb{N}$ is a small-step bisimulation up to context, by case analysis on t.

- If $t \rightarrow_{\mathrm{v}} t'$, then $\langle (\lambda x.\langle E[x]\rangle)\, t\rangle \rightarrow_{\mathrm{v}} \langle (\lambda x.\langle E[x]\rangle)\, t'\rangle$, $\langle E[t]\rangle \rightarrow_{\mathrm{v}} \langle E[t']\rangle$, and we have $\langle (\lambda x.\langle E[x]\rangle)\, t'\rangle \;\mathcal{R}\; \langle E[t']\rangle$, as required.

- If $t = v$, then $\langle (\lambda x.\langle E[x]\rangle)\, v\rangle \rightarrow_{\mathrm{v}} \langle\langle E[v]\rangle\rangle$. We have proved in Example 4.3 that $\langle\langle E[v]\rangle\rangle \;\mathbb{N}\; \langle E[v]\rangle$.

- If $t = F[yv]$, then we have to relate $\langle (\lambda x.\langle E[x]\rangle)\, F\rangle$ and $\langle E[F]\rangle$ (we already have $v \;\mathbb{N}^{\mathsf{NF}_{\mathrm{v}}}\; v$). If $F = F'[\langle E'\rangle]$, then we have $\langle (\lambda x.\langle E[x]\rangle)\, F'[z]\rangle \;\mathcal{R}\; \langle E[F'[z]]\rangle$ and $\langle E'[z]\rangle \;\mathbb{N}\; \langle E'[z]\rangle$ for a fresh z. If $F = E'$, then we have $\langle (\lambda x.\langle E[x]\rangle)\, E'[z]\rangle \;\mathcal{R}\; \langle E[E'[z]]\rangle$ for a fresh z.

- If $t = E'[\mathcal{S}k.t']$, then $\langle (\lambda x.\langle E[x]\rangle)\, t\rangle \rightarrow_{\mathrm{v}} \langle t'\{\lambda y.\langle (\lambda x.\langle E[x]\rangle)\, E'[y]\rangle/k\}\rangle$, and also $\langle E[t]\rangle \rightarrow_{\mathrm{v}} \langle t'\{\lambda y.\langle E[E'[y]]\rangle/k\}\rangle$. Because $\langle (\lambda x.\langle E[x]\rangle)\, E'[y]\rangle \;\mathcal{R}\; \langle E[E'[y]]\rangle$ holds, we deduce that we have $\langle t'\{\lambda y.\langle (\lambda x.\langle E[x]\rangle)\, E'[y]\rangle/k\}\rangle \;\widehat{\mathcal{R}}\; \langle t'\{\lambda y.\langle E[E'[y]]\rangle/k\}\rangle$, as wished.

Without using small-step bisimulation, the definition of $\mathcal{R}$ as well as the above bisimulation proof would be much more complex, since we would have to compute the results of the evaluations of $\langle (\lambda x.\langle E[x]\rangle)\, t\rangle$ and of $\langle E[t]\rangle$, which is particularly difficult if t is a control stuck term.

4.4. Refined Normal-Form Bisimilarity

We propose in this subsection a refinement of the definition of normal-form bisimilarity. Indeed, Definition 4.1 is too discriminating with control stuck terms, as we can see with the following terms.

Proposition 4.18. *Let* $i \stackrel{\mathsf{def}}{=} \lambda x.x$. *We have* $\mathcal{S}k.i \;\mathbb{C}\; (\mathcal{S}k.i)\Omega$, *but* $\mathcal{S}k.i \;\not\mathbb{N}\; (\mathcal{S}k.i)\Omega$.

Proof. We can easily prove that $\mathcal{S}k.i \;\mathbb{C}\; (\mathcal{S}k.i)\,\Omega$ holds with applicative bisimilarity or Definition 4.19. They are not normal-form bisimilar, since the contexts $\square$ and $\square\,\Omega$ are not related by $\mathbb{N}^{\mathsf{NF}\eta}$ (x converges while $x\,\Omega$ diverges).

When comparing control stuck terms, normal-form bisimilarity considers contexts and shift bodies separately, while they are combined if the control stuck terms are put under a reset and the capture goes through. To fix this issue, we consider another notion of bisimulation. Given a relation $\mathcal{R}$ on terms, we define $\mathcal{R}^{\mathsf{RNF}\eta}$ on normal forms, which changes the way $\mathcal{R}^{\mathsf{NF}\eta}$ operates on control stuck terms:

$$\frac{t_0 \;\mathcal{R}^{\mathsf{NF}\eta}\; t_1 \qquad t_0, t_1 \text{ not control stuck}}{t_0 \;\mathcal{R}^{\mathsf{RNF}\eta}\; t_1}$$

$$\frac{\langle t_0\{\lambda x.\langle k'\; E_0[x]\rangle/k\}\rangle \;\mathcal{R}\; \langle t_1\{\lambda x.\langle k'\; E_1[x]\rangle/k\}\rangle \qquad k', x \text{ fresh}}{E_0[\mathcal{S}k.t_0] \;\mathcal{R}^{\mathsf{RNF}\eta}\; E_1[\mathcal{S}k.t_1]}$$

The rule for control stuck terms simulates the capture of E_0 (respectively E_1) by $\mathcal{S}k.t_0$ (respectively $\mathcal{S}k.t_1$), but with an additional k'. Indeed, if $E_0[\mathcal{S}k.t_0]$ is put into a context $\langle E \rangle$, then $\mathcal{S}k.t_0$ captures a context bigger than E_0, namely $E[E_0]$. We take such possibility into account by introducing the variable k' in the definition of $\mathcal{R}^{\mathsf{RNF}\eta}$, which represents the context that can be captured beyond E_0 and E_1. Such a technique was introduced by Felleisen et al. to define a syntactic theory of control with local reduction rules [24, 25].

Definition 4.19. A relation $\mathcal{R}$ on terms is a refined normal-form simulation if $t_0 \; \mathcal{R} \; t_1$ and $t_0 \Downarrow_{\mathrm{v}} t'_0$ implies that there exists t'_1 such that $t_1 \Downarrow_{\mathrm{v}} t'_1$ and $t'_0 \; \mathcal{R}^{\mathsf{RNF}\eta} \; t'_1$. A relation $\mathcal{R}$ is a refined normal-form bisimulation if both $\mathcal{R}$ and $\mathcal{R}^{-1}$ are refined normal-form simulations. Refined normal-form bisimilarity, written $\mathbb{R}$, is the largest refined normal-form bisimulation.

Refined bisimilarity contains regular bisimilarity.

Proposition 4.20. *We have* $\mathbb{N} \subset \mathbb{R}$.

Indeed, for control stuck terms, if $t_0 \; \mathbb{N} \; t_1$ and $t_0 \Downarrow_{\mathrm{v}} E_0[\mathcal{S}k.t'_0]$, then $t_1 \Downarrow_{\mathrm{v}} E_1[\mathcal{S}k.t'_1]$, with $E_0 \; \mathbb{N}^{\mathsf{NF}\eta} \; E_1$, and $\langle t'_0 \rangle \; \mathbb{N} \; \langle t'_1 \rangle$. Because $\mathbb{N}$ is a congruence (Theorem 4.5) and substitutive, it is easy to see that

$$\langle t'_0\{\lambda x.\langle k' \; E_0[x]\rangle/k\}\rangle \; \mathbb{N} \; \langle t'_1\{\lambda x.\langle k' \; E_1[x]\rangle/k\}\rangle$$

holds for fresh k' and x. Therefore, $\mathbb{N}$ is a refined bisimulation, and is included in $\mathbb{R}$. The inclusion is strict, because $\mathbb{R}$ relates the terms of Proposition 4.18, while $\mathbb{N}$ does not.

Soundness. Proving that $\mathbb{R}$ is sound requires some adjustments to the congruence proof of $\mathbb{N}$. We introduce a special kind of substitution (called context substitution) $t\{E/k\}$, which replaces a variable in function position with a pure context. This operation is the same as the replacement of names with contexts in the $\lambda\mu$-calculus [65].

$$\begin{aligned}
x\{E/k\} &\stackrel{\mathsf{def}}{=} x \text{ if } x \neq k \\
(\lambda x.t)\{E/k\} &\stackrel{\mathsf{def}}{=} \lambda x.t\{E/k\} \text{ if } x \notin \mathsf{fv}(E) \cup \{k\} \\
(k\,t)\{E/k\} &\stackrel{\mathsf{def}}{=} E[t\{E/k\}] \\
(t_0\,t_1)\{E/k\} &\stackrel{\mathsf{def}}{=} t_0\{E/k\}\,t_1\{E/k\} \text{ if } t_0 \neq k \\
\langle t\rangle\{E/k\} &\stackrel{\mathsf{def}}{=} \langle t\{E/k\}\rangle \\
(\mathcal{S}k'.t)\{E/k\} &\stackrel{\mathsf{def}}{=} \mathcal{S}k'.t\{E/k\} \text{ if } k' \notin \mathsf{fv}(E) \cup \{k\}
\end{aligned}$$

The idea is to replace the fresh variables introduced in the control stuck terms case of Definition 4.19 (which are always in subterms of the form $\langle k\,t\rangle$) with a context when needed. We define a notion of well-formedness w.r.t. a variable k to characterize the terms to which context substitution can be applied.

Definition 4.21. A term t is well-formed w.r.t. k, written $\mathsf{wf}_k(t)$, if any free occurrence of k in t appears in a sub-term of the form $\langle k\,t'\rangle$ for some t'.

Proposition 4.22. *If* $\mathsf{wf}_k(t)$ *and* $t \rightarrow_{\mathrm{v}} t'$, *then* $\mathsf{wf}_k(t')$.

PROOF. By case analysis on the reduction rules.

Henceforth, when we write $t\{E/k\}$, we assume that $\mathsf{wf}_k(t)$ holds. To prove the congruence of $\mathbb{R}$, we proceed as in Section 4.2, and consider pairs of terms of the form $(\overrightarrow{F_0}[t_0]\overrightarrow{\sigma_0}, \overrightarrow{F_1}[t_1]\overrightarrow{\sigma_1})$, except now sequences of substitutions $\overrightarrow{\sigma}$ range over value and context substitutions. When these terms reduce to control stuck terms, we use context substitution to conclude. The complete congruence proof can be found in Appendix A.3.

Theorem 4.23. *The relation* $\mathbb{R}$ *is a congruence.*

Theorem 4.24. *We have* $\mathbb{R} \subset \mathbb{C}$.

The inclusion is strict, because the terms of Proposition 4.7 are still not related by $\mathbb{R}$. We would like to stress that even though $\mathbb{R}$ equates more contextually equivalent terms than $\mathbb{N}$, the latter is still useful, since it leads to very simple proofs of equivalence, as we can see, e.g., with the examples of Section 4.5. Therefore, $\mathbb{R}$ does not disqualify $\mathbb{N}$ as a proof technique.

Example 4.25. We prove that if $k' \notin \mathsf{fv}(E) \cup \mathsf{fv}(t)$ and $x \notin \mathsf{fv}(E)$, then $E[\mathcal{S}k.t] \mathrel{\mathbb{R}} \mathcal{S}k'.t\{\lambda x.\langle k'\; E[x]\rangle/k\}$. The two terms are control stuck terms, therefore, we have to prove that $\langle t\{\lambda x.\langle k''\; E[x]\rangle/k\}\rangle \mathrel{\mathbb{R}} \langle t\{\lambda x.\langle(\lambda y.\langle k''\; y\rangle)\; E[x]\rangle/k\}\rangle$ holds for a fresh k''. We know that $\langle k''\; E[x]\rangle \mathrel{\mathbb{N}} \langle(\lambda y.\langle k''\; y\rangle)\; E[x]\rangle$ holds by Example 4.17. Consequently, we have $\langle k''\; E[x]\rangle \mathrel{\mathbb{R}} \langle(\lambda y.\langle k''\; y\rangle)\; E[x]\rangle$ by Proposition 4.20. We can then conclude by congruence of $\mathbb{R}$.

Proving this result with the regular normal-form bisimulation would require us to equate $E[y]$ and y (where y is fresh), which is clearly not true in general (take, for instance, $E = (\lambda z.\Omega)\; \square$).

Bisimulation up to context. We can also define a notion of bisimulation up to context sound w.r.t. refined bisimilarity. As in the regular case, our proof technique fails with a refined up-to technique which respects η-expansion. Given a relation $\mathcal{R}$ on terms, we, therefore, define $\mathcal{R}^{\mathsf{RNF}\searrow}$ as follows

$$\frac{t_0 \mathrel{\mathcal{R}^{\mathsf{NF}\searrow}} t_1 \qquad t_0, t_1 \text{ not control stuck}}{t_0 \mathrel{\mathcal{R}^{\mathsf{RNF}\searrow}} t_1}$$

$$\frac{\langle t_0\{\lambda x.\langle k'\; E_0[x]\rangle/k\}\rangle \mathrel{\mathcal{R}} \langle t_1\{\lambda x.\langle k'\; E_1[x]\rangle/k\}\rangle \qquad k', x \text{ fresh}}{E_0[\mathcal{S}k.t_0] \mathrel{\mathcal{R}^{\mathsf{RNF}\searrow}} E_1[\mathcal{S}k.t_1]}$$

We can then define refined non-η bisimulation up to reduction as in Definition 4.11, replacing $\mathcal{R}^{\mathsf{NF}\searrow}$ with $\mathcal{R}^{\mathsf{RNF}\searrow}$. Finally, we define the closure $\breve{\mathcal{R}}$ by the rules in Figure 5, which are basically the same as in Figure 4, with an extra rule for context substitution. In this rule, we relate the contexts E_0, E_1 within an enclosing reset, because we substitute them for variables that are in subterms of the form $\langle k\, t\rangle$; consequently, we know that they will be executed with a surrounding reset. We then define refined bisimulation up to context as follows.

$$\frac{}{t \mathrel{\breve{\mathcal{R}}} t} \qquad \frac{t_0 \mathrel{\mathcal{R}} t_1}{t_0 \mathrel{\breve{\mathcal{R}}} t_1} \qquad \frac{t_0 \mathrel{\breve{\mathcal{R}}} t_1 \qquad v_0 \mathrel{\breve{\mathcal{R}}^{\mathsf{RNF}\searrow}} v_1}{t_0\{v_0/x\} \mathrel{\breve{\mathcal{R}}} t_1\{v_1/x\}}$$

$$\frac{t_0 \mathrel{\breve{\mathcal{R}}} t_1 \qquad \langle E_0 \rangle \mathrel{\breve{\mathcal{R}}^{\mathsf{C}\searrow}} \langle E_1 \rangle}{t_0\{E_0/k\} \mathrel{\breve{\mathcal{R}}} t_1\{E_1/k\}} \qquad \frac{t_0 \mathrel{\breve{\mathcal{R}}} t_1 \qquad F_0 \mathrel{\breve{\mathcal{R}}^{\mathsf{C}\searrow}} F_1}{F_0[t_0] \mathrel{\breve{\mathcal{R}}} F_1[t_1]}$$

$$\frac{t_0 \mathrel{\breve{\mathcal{R}}} t_1 \qquad t_0, t_1 \text{ delimited} \qquad F_0[x] \mathrel{\breve{\mathcal{R}}^{\searrow}} F_1[x] \qquad x \text{ fresh}}{F_0[t_0] \mathrel{\breve{\mathcal{R}}} F_1[t_1]}$$

$$\frac{t_0 \mathrel{\breve{\mathcal{R}}} t_1}{\lambda x.t_0 \mathrel{\breve{\mathcal{R}}} \lambda x.t_1} \qquad \frac{t_0 \mathrel{\breve{\mathcal{R}}} t_1}{\mathcal{S}k.t_0 \mathrel{\breve{\mathcal{R}}} \mathcal{S}k.t_1}$$

Figure 5: Refined closure of a relation $\mathcal{R}$

Definition 4.26. A relation $\mathcal{R}$ on terms is a refined normal-form simulation up to context if $t_0 \mathrel{\mathcal{R}} t_1$ and $t_0 \Downarrow_{\mathrm{v}} t_0'$ implies that there exists t_1' such that $t_1 \Downarrow_{\mathrm{v}} t_1'$ and $t_0' \mathrel{\breve{\mathcal{R}}^{\mathsf{RNF}\searrow}} t_1'$. A relation $\mathcal{R}$ is a refined normal-form bisimulation up to context if both $\mathcal{R}$ and $\mathcal{R}^{-1}$ are refined normal-form simulations up to context.

We can adapt the proof of Proposition 4.13 to prove the soundness of refined bisimulation up to context w.r.t. $\mathbb{R}$.

Proposition 4.27. *If $\mathcal{R}$ is a refined bisimulation up to context, then $\breve{\mathcal{R}}$ is a refined non-η bisimulation.*

In the cases featuring control stuck terms, we rely on context substitution to conclude (see Appendix A.3 for more details).

4.5. Proving the Axioms

We now show how the axioms can be proved using normal-form bisimulation. Because we have so far been working with the relaxed semantics, we remind that the $\mathcal{S}_{\mathtt{elim}}$ axiom does not hold, as discussed in Section 3.1. We show how to prove the $\mathcal{S}_{\mathtt{elim}}$ axiom with the delimited normal-form bisimulation of the next section. The η-equivalence axiom ($\eta_{\mathtt{v}}$ axiom) holds by definition of $\mathbb{N}^{\mathsf{NF}\eta}$.

Proposition 4.28 ($\beta_{\mathtt{v}}$, $\langle\cdot\rangle_{\mathcal{S}}$, and $\langle\cdot\rangle_{\mathtt{val}}$ axioms). *We have $(\lambda x.t)\, v \mathrel{\mathbb{N}} t\{v/x\}$, $\langle E[\mathcal{S}k.t]\rangle \mathrel{\mathbb{N}} \langle t\{\lambda x.\langle E[x]\rangle/k\}\rangle$ (x fresh), and $\langle v \rangle \mathrel{\mathbb{N}} v$.*

Proof. These are direct consequences of Proposition 4.2.

Proposition 4.29 ($\mathcal{S}_{\langle\cdot\rangle}$ axiom). *We have $\mathcal{S}k.\langle t\rangle \mathrel{\mathbb{N}} \mathcal{S}k.t$.*

PROOF. We want to relate two stuck terms, so using normal-form bisimulation, we have to show $\langle\langle t\rangle\rangle \ \mathbb{N}\ \langle t\rangle$ (proved in Example 4.3) and $\square\ \mathbb{N}^{\mathsf{C}}\ \square$ (a consequence of the fact that $\mathbb{N}$ is reflexive).

Proposition 4.30 ($\langle\cdot\rangle_{\texttt{lift}}$ axiom). *We have* $\langle(\lambda x.t_0)\ \langle t_1\rangle\rangle \ \mathbb{N}\ (\lambda x.\langle t_0\rangle)\ \langle t_1\rangle$.

PROOF. We prove that $\mathcal{R} \overset{\mathsf{def}}{=} \{(\langle(\lambda x.t_0)\ \langle t_1\rangle\rangle, (\lambda x.\langle t_0\rangle)\ \langle t_1\rangle) \mid (t_0, t_1) \in \mathcal{T}^2\} \cup \{(t,t) \mid t \in \mathcal{T}\}$ is a normal-form bisimulation. The terms $\langle(\lambda x.t_0)\ \langle t_1\rangle\rangle$ and $(\lambda x.\langle t_0\rangle)\ \langle t_1\rangle$ reduce to a normal form iff $\langle t_1\rangle$ reduces to a normal form, and according to Proposition 2.9, we have two cases.

If $\langle t_1\rangle \Downarrow_{\mathrm{v}} v$, then $\langle(\lambda x.t_0)\ \langle t_1\rangle\rangle \rightarrow_{\mathrm{v}}^* \langle t_0\{v/x\}\rangle$ and $(\lambda x.\langle t_0\rangle)\langle t_1\rangle \rightarrow_{\mathrm{v}}^* \langle t_0\{v/x\}\rangle$. Therefore, $\langle(\lambda x.t_0)\ \langle t_1\rangle\rangle \Downarrow_{\mathrm{v}} t''$ iff $(\lambda x.\langle t_0\rangle)\ \langle t_1\rangle \Downarrow_{\mathrm{v}} t''$, and we have $t''\ \mathcal{R}^{\mathsf{NF}\eta}\ t''$, as required.

If $\langle t_1\rangle$ reduces to an open stuck term, then $\langle t_1\rangle \Downarrow_{\mathrm{v}} \langle F[y\,v]\rangle$ by Proposition 2.9. In this case, we have $\langle(\lambda x.t_0)\ \langle t_1\rangle\rangle \Downarrow_{\mathrm{v}} \langle(\lambda x.t_0)\ \langle F[y\,v]\rangle\rangle$ and also $(\lambda x.\langle t_0\rangle)\ \langle t_1\rangle \Downarrow_{\mathrm{v}} (\lambda x.\langle t_0\rangle)\ \langle F[y\,v]\rangle$. We have $v\ \mathcal{R}^{\mathsf{NF}\eta}\ v$, and it is easy to check that $\langle(\lambda x.t_0)\ \langle F\rangle\rangle\ \mathcal{R}^{\mathsf{C}}\ (\lambda x.\langle t_0\rangle)\ \langle F\rangle$ holds, whether F is pure or not.

Proposition 4.31 (β_Ω axiom). *If $x \notin \mathsf{fv}(E)$, then $(\lambda x.E[x])\ t\ \mathbb{N}\ E[t]$.*

PROOF. We prove that $\mathcal{R} \overset{\mathsf{def}}{=} \{((\lambda x.E[x])\ t, E[t]) \mid t \in \mathcal{T}, E \in \mathcal{PC}, x \notin \mathsf{fv}(E)\} \cup \{(t,t) \mid t \in \mathcal{T}\}$ is a normal-form bisimulation. If $(\lambda x.E[x])\ t$ evaluates to some normal form, then t evaluates to some normal form as well. We distinguish three cases. If $t \Downarrow_{\mathrm{v}} v$, then $(\lambda x.E[x])\ t \rightarrow_{\mathrm{v}}^* E[v]$ (because $x \notin \mathsf{fv}(E)$), and $E[t] \rightarrow_{\mathrm{v}}^* E[v]$. We obtain the same term in both cases, and from there, it is easy to conclude.

If $t \Downarrow_{\mathrm{v}} F[y v]$, then $(\lambda x.E[x])\, t \Downarrow_{\mathrm{v}} (\lambda x.E[x])\, F[y v]$, and $E[t] \Downarrow_{\mathrm{v}} E[F[x\, v]]$. We have to prove $v\ \mathcal{R}^{\mathsf{NF}\eta}\ v$, which is obvious, and $(\lambda x.E[x])\ F\ \mathcal{R}^{\mathsf{C}}\ E[F]$. Let z be a fresh variable. If F is a pure context E', we have to prove $(\lambda x.E[x])\ E'[z]\ \mathcal{R}\ E[E'[z]]$, which is clearly true. Otherwise $F = F'[\langle E'\rangle]$, and we have to prove $(\lambda x.E[x])\, F'[z]\ \mathcal{R}\ E[F'[z]]$, which is clearly true, and $\langle E'[z]\rangle\ \mathcal{R}\ \langle E'[z]\rangle$, which is true as well because $\mathcal{R}$ contains the identity relation.

If $t \Downarrow_{\mathrm{v}} E'[\mathcal{S}k.t']$, then we have $(\lambda x.E[x])\, t \Downarrow_{\mathrm{v}} (\lambda x.E[x])\, E'[\mathcal{S}k.t']$, and $E[t] \Downarrow_{\mathrm{v}} E[E'[\mathcal{S}k.t']]$. Let y be a fresh variable. We have to prove $(\lambda x.E[x])\ E'[y]\ \mathcal{R}\ E[E'[y]]$, which is clearly true, and $\langle t'\rangle\ \mathcal{R}\ \langle t'\rangle$, which is true as well.

4.6. Normal-Form Bisimulation for the Original Semantics

The relations defined so far are tailored for the relaxed semantics. We now propose a normal-form bisimulation for the original semantics, where a control stuck term can be equivalent to a term which does not evaluate to a stuck term. To do so, a simple idea would be to relate two terms t_0 and t_1 that are not delimited by comparing $\langle t_0\rangle$ and $\langle t_1\rangle$: any potentially control stuck term would be "unstuck" by the surrounding reset. However, such a solution would not be sound, as it would relate $\mathcal{S}k.k\ y$ and $\mathcal{S}k.(\lambda z.z)\ y$, terms that can be distinguished by the context $\langle\square\ \Omega\rangle$.

To take into account the fact that t_0 and t_1 can be put in a context before being surrounded by a reset, we use the same technique as for refined bisimulation (Section 4.4): we compare $\langle k\, t_0\rangle$ and $\langle k\, t_1\rangle$, where k is a fresh variable which stands for a potential (pure) evaluation context. As a result, we define *delimited normal-form bisimulation* as follows.

Definition 4.32. A relation $\mathcal{R}$ on terms is a delimited normal-form simulation if $t_0 \mathrel{\mathcal{R}} t_1$ implies:

- if t_0, t_1 are not both delimited, then $\langle k\, t_0\rangle \mathrel{\mathcal{R}} \langle k\, t_1\rangle$ holds for a fresh k;
- otherwise, if $t_0 \Downarrow_{\mathrm{v}} t_0'$, then there exists t_1' such that $t_1 \Downarrow_{\mathrm{v}} t_1'$ and $t_0' \mathrel{\mathcal{R}^{\mathsf{NF}\eta}} t_1'$.

A relation $\mathcal{R}$ is a delimited normal-form bisimulation if both $\mathcal{R}$ and $\mathcal{R}^{-1}$ are delimited normal-form simulations. Delimited normal-form bisimilarity, written $\mathbb{M}$, is the largest delimited normal-form bisimulation.

Pairs of delimited terms are treated as in Definition 4.1, by reducing them to normal forms, which are then compared using $\mathcal{R}^{\mathsf{NF}\eta}$. Note that delimited terms cannot reduce to control stuck terms (Proposition 2.9), therefore, the control stuck term case of the definition of $\mathcal{R}^{\mathsf{NF}\eta}$ becomes useless with delimited bisimulation. So, in fact, we use the restriction of $\mathcal{R}^{\mathsf{NF}\eta}$ to values and open stuck terms in Definition 4.32.

Example 4.33 (double reset). The terms $\langle\langle t\rangle\rangle$ and $\langle t\rangle$ are delimited, therefore, we can prove they are delimited bisimilar by using the same bisimulation as in Example 4.3. In contrast, to equate $\mathcal{S}k.t$ and $\mathcal{S}k.\langle t\rangle$ ($\mathcal{S}_{\langle\cdot\rangle}$ axiom), we have to relate $\langle k'\, \mathcal{S}k.t\rangle$ and $\langle k'\, \mathcal{S}k.\langle t\rangle\rangle$ for a fresh k'. These two terms reduce to, respectively, $\langle t\{\lambda x.\langle k'\, x\rangle/k\}\rangle$ and $\langle\langle t\{\lambda x.\langle k'\, x\rangle/k\}\rangle\rangle$ – two terms of the form $\langle t'\rangle$ and $\langle\langle t'\rangle\rangle$ for some t'; they are, therefore, delimited bisimilar.

Actually, the regular and refined normal-form bisimulations can be used as proof techniques for delimited bisimilarity.

Proposition 4.34. *We have* $\mathbb{N} \subseteq \mathbb{M}$ *and* $\mathbb{R} \subseteq \mathbb{M}$.

Proof. We prove that $\mathbb{N}$ is a delimited normal-form bisimulation. For terms $t_0 \mathrel{\mathbb{N}} t_1$ which are both delimited, the definitions of $\mathbb{N}$ and $\mathbb{M}$ coincide. If $t_0 \mathrel{\mathbb{N}} t_1$ with t_0 or t_1 not delimited, we have to prove that $\langle k\, t_0\rangle \mathrel{\mathbb{N}} \langle k\, t_1\rangle$ holds, which is true because $\mathbb{N}$ is a congruence. The reasoning is the same for $\mathbb{R}$.

As a result, any terms already proved normal-form or refined bisimilar are delimited bisimilar, like, e.g., the axioms (Section 4.5). In particular, we have $\rightarrow_{\mathrm{v}} \subseteq \mathbb{M}$ (and, therefore, $\Downarrow_{\mathrm{v}} \subseteq \mathbb{M}$).

Soundness. We prove that $\mathbb{M}$ is a congruence in two steps: first, we show congruence on delimited terms, from which we can deduce congruence for all terms. For delimited terms, we proceed as in Sections 4.2 and 4.4, by considering terms of the form $(\overrightarrow{F_0}[t_0]\overrightarrow{\sigma_0}, \overrightarrow{F_1}[t_1]\overrightarrow{\sigma_1})$, where t_0 and t_1 are delimited. As with refined bisimilarity, the above sequences of substitutions $\overrightarrow{\sigma}$ contain value and context substitutions.

Because delimited bisimilarity compares two non-delimited terms t_0, t_1 by turning them into delimited terms $\langle k\, t_0\rangle$, $\langle k\, t_1\rangle$, we can deduce compatibility w.r.t. evaluation contexts (application, reset) for all terms by considering $\langle k\, t_0\rangle$ and $\langle k\, t_1\rangle$, and then substitute the appropriate contexts for k. Compatibility w.r.t. shift and λ-abstraction can then easily be proved separately. The complete proofs can be found in Appendix A.4. We deduce that $\mathbb{M}$ is sound w.r.t. $\mathbb{P}$.

Theorem 4.35. *We have* $\mathbb{M} \subseteq \mathbb{P}$*.*

However, $\mathbb{M}$ is not complete, since the terms of Proposition 4.7 are still related by $\mathbb{P}$ but not by $\mathbb{M}$.

Bisimulation up to context. A notion of bisimulation up to context can be defined for the original semantics, which does not respect η-expansion (as in the relaxed semantics). Given a relation $\mathcal{R}$, we consider $\mathcal{R}^{\mathsf{NF}\searrow}$ without its clause for control stuck terms, and we use the same closure $\breve{\mathcal{R}}$ as with refined bisimulation up to context (Section 4.4); we remind that $\breve{\mathcal{R}}$ contains a clause for context substitution. We define delimited bisimulation up to context as follows.

Definition 4.36. A relation $\mathcal{R}$ on terms is a delimited normal-form simulation up to context if $t_0 \mathrel{\mathcal{R}} t_1$ implies:

- if t_0, t_1 are not both delimited, then $\langle k\, t_0\rangle \mathrel{\mathcal{R}} \langle k\, t_1\rangle$ holds for a fresh k;
- otherwise, if $t_0 \Downarrow_{\mathrm{v}} t_0'$, then there exists t_1' such that $t_1 \Downarrow_{\mathrm{v}} t_1'$ and $t_0' \mathrel{\breve{\mathcal{R}}^{\mathsf{NF}\searrow}} t_1'$.

A relation $\mathcal{R}$ is a delimited normal-form bisimulation up to context if both $\mathcal{R}$ and $\mathcal{R}^{-1}$ are delimited normal-form simulations up to context.

It would not be sound to have $\langle k\, t_0\rangle \mathrel{\breve{\mathcal{R}}} \langle k\, t_1\rangle$ in the first item, because $t_0 \mathrel{\mathcal{R}} t_1$ implies $\langle k\, t_0\rangle \mathrel{\breve{\mathcal{R}}} \langle k\, t_1\rangle$ for any relation $\mathcal{R}$. We prove that delimited bisimulation up to context is sound the same way as for the relaxed semantics, by defining a notion of delimited non-η bisimulation up to reduction, and we show the following result (see Appendix A.5).

Lemma 4.37. *Let $\mathcal{R}$ be a delimited bisimulation up to context. Let $t_0 \mathrel{\breve{\mathcal{R}}} t_1$.*

1. *If t_0, t_1 are not both delimited, then for all fresh k, if $\langle k\, t_0\rangle \Downarrow_{\mathrm{v}} t_0'$, there exists t_1' such that $\langle k\, t_1\rangle \Downarrow_{\mathrm{v}} t_1'$ and $t_0' \mathrel{\breve{\mathcal{R}}^{\mathsf{NF}\searrow}} t_1'$ (and conversely if $\langle k\, t_1\rangle \Downarrow_{\mathrm{v}} t_1'$) .*
2. *If t_0, t_1 are both delimited and $t_0 \Downarrow_{\mathrm{v}} t_0'$, there exists t_1' such that $t_1 \Downarrow_{\mathrm{v}} t_1'$ and $t_0' \mathrel{\breve{\mathcal{R}}^{\mathsf{NF}\searrow}} t_1'$ (and conversely if $t_1 \Downarrow_{\mathrm{v}} t_1'$).*

This proves that if $\mathcal{R}$ is a delimited bisimulation up to context, then $\breve{\mathcal{R}}$ is a delimited non-η bisimulation up to reduction, and is therefore contained in $\mathbb{M}$. We can also define a small-step delimited bisimulation up to context and prove its soundness the same way. In the next section, we use delimited bisimulation up to context to prove the $\mathcal{S}_{\mathtt{elim}}$ axiom.

4.7. Examples

We illustrate the differences between the equivalences of the relaxed and original semantics by giving some examples of terms related by $\mathbb{M}$ (and, therefore, by $\mathbb{P}$), but not by the contextual equivalence of the relaxed semantics $\mathbb{C}$. First, note that $\mathbb{M}$ relates non-terminating terms with stuck non-terminating terms.

Proposition 4.38. *We have* $\Omega \mathrel{\mathbb{M}} \mathcal{S}k.\Omega$*.*

The relation $\{(\Omega, \mathcal{S}k.\Omega), (\langle k'\, \Omega\rangle, \langle k'\, \mathcal{S}k.\Omega\rangle)\}$ (where $k' \neq k$) is a delimited bisimulation. Proposition 4.38 does not hold with $\mathbb{C}$ because Ω is not stuck.

As wished, $\mathbb{M}$ satisfies the only axiom of [38] not satisfied by $\mathbb{C}$.

Proposition 4.39 ($\mathcal{S}_{\mathtt{elim}}$ axiom). *If* $k \notin \mathsf{fv}(t)$*, then* $t \mathrel{\mathbb{M}} \mathcal{S}k.k\, t$*.*

Proof. The relation

$$\{(t, \mathcal{S}k.k\,t), (\langle k'\, t\rangle, \langle k'\, \mathcal{S}k.k\,t\rangle), (\langle k'\, t\rangle, \langle(\lambda x.\langle k'\, x\rangle)\, t\rangle) \\ \mid t \in \mathcal{T}, \{k, k', x\} \cap \mathsf{fv}(t) = \emptyset\} \cup \mathbb{M}$$

is a small-step delimited bisimulation up to context. We have $\langle k'\, \mathcal{S}k.k\,t\rangle \rightarrow_{\mathrm{v}} \langle(\lambda x.\langle k'\, x\rangle)\, t\rangle$, and we can relate $\langle k'\, t\rangle$ and $\langle(\lambda x.\langle k'\, x\rangle)\, t\rangle$ with up-to context as in Example 4.17.

Consequently, $\mathbb{M}$ is complete w.r.t. $\equiv$.

Corollary 4.40. *We have* $\equiv\, \subseteq \mathbb{M}$*.*

As a result, we can use $\equiv$ as a proof technique for $\mathbb{M}$ (and, therefore, for $\mathbb{P}$). E.g., the following equivalence can be derived from the axioms [38].

Proposition 4.41. *If* $k \notin \mathsf{fv}(t_1)$*, then* $(\lambda x.\mathcal{S}k.t_0)\, t_1 \mathrel{\mathbb{M}} \mathcal{S}k.((\lambda x.t_0)\, t_1)$*.*

This equivalence does not hold with $\mathbb{C}$, because the term on the right is stuck, but the term on the left may not evaluate to a stuck term (if t_1 does not terminate). A direct proof of Proposition 4.41 (which does not rely on $\equiv$) is given below, as a use case for bisimulation up to context.

Proof. We want to relate $\langle k'\, ((\lambda x.\mathcal{S}k.t_0)\, t_1)\rangle$ with $\langle k'\, \mathcal{S}k.((\lambda x.t_0)\, t_1)\rangle$ for a fresh k'. We have $\langle k'\, \mathcal{S}k.((\lambda x.t_0)\, t_1)\rangle \rightarrow_{\mathrm{v}} \langle(\lambda x.t_0\{\lambda y.\langle k'\, y\rangle/k\})\, t_1\rangle$, therefore, we prove that $\mathcal{R} \stackrel{\mathsf{def}}{=} \{(\langle k'\, ((\lambda x.\mathcal{S}k.t_0)\, t_1)\rangle, \langle(\lambda x.t_0\{\lambda y.\langle k'\, y\rangle/k\})\, t_1\rangle) \mid (t_0, t_1) \in \mathcal{T}^2, k \notin \mathsf{fv}(t_1), \{k', y\} \cap \mathsf{fv}(t_0, t_1) = \emptyset\} \cup \mathbb{M}$ is a small-step bisimulation up to context, by case analysis on t_1.

- If $t_1 \rightarrow_{\mathrm{v}} t_1'$, then we have $\langle k' \; ((\lambda x.\mathcal{S}k.t_0) \; t_1)\rangle \rightarrow_{\mathrm{v}} \langle k' \; ((\lambda x.\mathcal{S}k.t_0) \; t_1')\rangle$ and $\langle (\lambda x.t_0\{\lambda y.\langle k' \; y\rangle/k\}) \; t_1\rangle \rightarrow_{\mathrm{v}} \langle (\lambda x.t_0\{\lambda y.\langle k' \; y\rangle/k\}) \; t_1'\rangle$, therefore, we obtain two terms in $\mathcal{R}$.

- If $t_1 = v_1$, then we have $\langle k' \; ((\lambda x.\mathcal{S}k.t_0) \; v_1)\rangle \rightarrow_{\mathrm{v}}^2 \langle t_0\{\lambda y.\langle k' \; y\rangle/k\}\{v_1/x\}\rangle$ and $\langle (\lambda x.t_0\{\lambda y.\langle k' \; y\rangle/k\}) \; t_1\rangle \rightarrow_{\mathrm{v}} \langle t_0\{\lambda y.\langle k' \; y\rangle/k\}\{v_1/x\}\rangle$; we obtain two identical terms.

- If $t_1 = F_1[y \; v_1]$, then we want to relate the contexts $\langle k' \; ((\lambda x.\mathcal{S}k.t_0) \; F_1)\rangle$ and $\langle (\lambda x.t_0\{\lambda y.\langle k' \; y\rangle/k\}) \; F_1\rangle$ (we already have $v_1 \; \mathbb{M}^{\mathsf{NF}_{\searrow}} \; v_1$). If $F_1 = F_1'[\langle E_1'\rangle]$, then $\langle k' \; ((\lambda x.\mathcal{S}k.t_0) \; F_1'[z])\rangle \; \mathcal{R} \; \langle (\lambda x.t_0\{\lambda y.\langle k' \; y\rangle/k\}) \; F_1'[z]\rangle$ and $\langle E_1'[z]\rangle \; \mathbb{M} \; \langle E_1'[z]\rangle$ hold for a fresh z. If $F_1 = E_1'$, then we also have $\langle k' \; ((\lambda x.\mathcal{S}k.t_0) \; E_1'[z])\rangle \; \mathcal{R} \; \langle (\lambda x.t_0\{\lambda y.\langle k' \; y\rangle/k\}) \; E_1'[z]\rangle$ for a fresh z.

- If $t_1 = E_1[\mathcal{S}k''.t_1']$, then

$$\langle k' \; ((\lambda x.\mathcal{S}k.t_0) \; t_1)\rangle \rightarrow_{\mathrm{v}} \langle t_1'\{\lambda z.\langle k' \; ((\lambda x.\mathcal{S}k.t_0) \; E_1[z])\rangle/k''\}\rangle \text{ and}$$
$$\langle (\lambda x.t_0\{\lambda y.\langle k' \; y\rangle/k\}) \; t_1\rangle \rightarrow_{\mathrm{v}} \langle t_1'\{\lambda z.\langle (\lambda x.t_0\{\lambda y.\langle k' \; y\rangle/k\}) \; E_1[z]\rangle/k''\}\rangle$$

  Because we have $\langle k' \; ((\lambda x.\mathcal{S}k.t_0) \; E_1[z])\rangle \; \mathcal{R} \; \langle (\lambda x.t_0\{\lambda y.\langle k' \; y\rangle/k\}) \; E_1[z]\rangle$, we obtain two terms in $\breve{\mathcal{R}}$, as wished.

Finally, there exist also terminating terms in $\mathbb{P}$ but not in $\mathbb{C}$ which are not CPS equivalent. For example, if we combine Turing and Curry fixed-point combinators (see Example 2.2) with the $\mathcal{S}_{\mathtt{elim}}$ axiom, we obtain the following result.

Proposition 4.42. *We have $\mathcal{S}k.k \; \Theta \; \mathbb{M} \; \Delta$.*

The bisimulation proof is the same as for Proposition 4.39.

4.8. Conclusion

In this section, we propose several definitions of normal-form bisimilarities as well as up-to techniques for the relaxed and original semantics of $\lambda_{\mathcal{S}}$. The first definition of normal-form bisimulation (Definition 4.1) separates the contexts and bodies of shift operators when testing control stuck terms, while refined bisimulation (Definition 4.19) consider the contexts and bodies combined. Building a candidate relation to prove the equivalence of two given terms and proving it is a bisimulation is easier to do with the first definition compared to the refined one, but the refined one relates strictly more terms.

The bisimulation for the original semantics (Definition 4.32) has to be able to relate a control stuck term with a term which reduces to, e.g., a value, as in the $\mathcal{S}_{\mathtt{elim}}$ axiom. To do so, we transform the pairs where at least one term is not delimited into pairs of delimited terms, by putting them inside a context $\langle k \; \square\rangle$, where k is a fresh variable. We then treat pairs of delimited terms as before, except these terms cannot reduce to control stuck terms. In Section 4.7, we illustrate the differences between the relaxed and the original semantics by providing examples of terms that are equivalent with the original semantics but not with the relaxed one, like, in particular, the $\mathcal{S}_{\mathtt{elim}}$ axiom.

In the relaxed as well as in the original semantics, the proof obligations can be simplified further by using up-to techniques, in particular bisimulation up to context. We can see that the resulting equivalence proofs are quite simple, by looking at the proofs of the examples throughout the section. However, the bisimilarities are not complete w.r.t. contextual equivalence $\mathbb{C}$ (respectively $\mathbb{P}$), as witnessed with the terms of Proposition 4.7.

5. Applicative Bisimilarity

Applicative bisimilarity has been originally defined for the lazy λ-calculus [1]. The main idea is to reduce (closed) terms to values, and then compare the resulting λ-abstractions by applying them to an arbitrary argument. Applicative bisimilarity for deterministic languages is usually sound and complete (see, e.g., [30, 87, 29]); in particular, proving soundness relies on a systematic proof technique called Howe's method [34, 29]. However, the definition of up-to techniques remains an open issue for applicative bisimilarity.

Very few works have proposed a definition of applicative bisimilarity in a calculus with control. In [61], Merro and Biasi define an applicative bisimilarity which characterizes contextual equivalence in the *CPS calculus* [86], a minimal calculus which models the control features of functional languages with imperative jumps. In the $\lambda\mu$-calculus, Lassen [50] defines a sound but not complete applicative bisimilarity in call-by-name. In a recent work, we improve this result by defining a sound and complete applicative bisimilarity in both call-by-name and call-by-value [14]. In this section, we define a sound and complete applicative bisimilarity for the relaxed semantics of $\lambda_{\mathcal{S}}$. Our definition of applicative bisimilarity relies on a labeled transition system, introduced first. We then define the relation itself, prove its soundness and completeness, before showing how it can be used on some examples.

Unlike with normal-form bisimilarity, we work primarily with closed terms, and we then extend the equivalence definitions to open terms with open extension (Definition 3.5). Besides, this section deals only with the relaxed semantics. We conjecture an applicative bisimilarity can also be defined for the original semantics, but we do not know how to prove it sound. We discuss this issue in the next section (see Remark 6.37).

5.1. Labeled Transition System

One possible way to define an applicative bisimilarity is to rely on a labeled transition system (LTS), where the possible interactions of a term with its environment are encoded in the labels (see, e.g., [30, 29]). Using a LTS simplifies the definition of the bisimilarity and makes it easier to use some techniques in proofs, such as diagram chasing. In Figure 6, we define a LTS $t_0 \xrightarrow{\alpha} t_1$ with three kinds of transitions, where we assume all the terms to be closed. An *internal action* $t \xrightarrow{\tau} t'$ is an evolution from t to t' without any help from the surrounding

$$\frac{}{(\lambda x.t)\ v \xrightarrow{\tau} t\{v/x\}}\ (\beta_v) \qquad \frac{t_0 \xrightarrow{\tau} t_0'}{t_0\ t_1 \xrightarrow{\tau} t_0'\ t_1}\ (\mathit{left}_\tau) \qquad \frac{t \xrightarrow{\tau} t'}{v\ t \xrightarrow{\tau} v\ t'}\ (\mathit{right}_\tau)$$

$$\frac{}{\langle v\rangle \xrightarrow{\tau} v}\ (\mathit{reset}) \qquad \frac{t \xrightarrow{\tau} t'}{\langle t\rangle \xrightarrow{\tau} \langle t'\rangle}\ (\langle\cdot\rangle_\tau) \qquad \frac{t \xrightarrow{\Box} t'}{\langle t\rangle \xrightarrow{\tau} t'}\ (\langle\cdot\rangle_{\mathcal{S}})$$

$$\frac{}{\lambda x.t \xrightarrow{v} t\{v/x\}}\ (\mathit{val}) \qquad \frac{x \notin \mathsf{fv}(E)}{\mathcal{S}k.t \xrightarrow{E} \langle t\{\lambda x.\langle E[x]\rangle/k\}\rangle}\ (\mathit{shift})$$

$$\frac{t_0 \xrightarrow{E[\Box\ t_1]} t_0'}{t_0\ t_1 \xrightarrow{E} t_0'}\ (\mathit{left}_{\mathcal{S}}) \qquad \frac{t \xrightarrow{E[v\ \Box]} t'}{v\ t \xrightarrow{E} t'}\ (\mathit{right}_{\mathcal{S}})$$

Figure 6: Labeled Transition System

context; it corresponds to a reduction step from t to t'. The transition $v_0 \xrightarrow{v_1} t$ expresses the fact that v_0 needs to be applied to another value v_1 to evolve, reducing to t. Finally, the transition $t \xrightarrow{E} t'$ means that t is control stuck, and when t is put in a context E enclosed in a reset, the capture can be triggered, the result of which being t'. We do not have a case for open stuck terms, because we work with closed terms only.

Most rules for internal actions (Figure 6) are straightforward; the rules (β_v) and (*reset*) mimic the corresponding reduction rules, and the compositional rules (right_τ), (left_τ), and $(\langle\cdot\rangle_\tau)$ allow internal actions to happen within any evaluation context. The rule $(\langle\cdot\rangle_{\mathcal{S}})$ for context capture is explained later. Rule (*val*) defines the only possible transition for values. Note that while both rules (β_v) and (*val*) encode β-reduction, they are quite different in nature; in the former, the term $(\lambda x.t)\ v$ can evolve by itself, without any help from the surrounding context, while the latter expresses the possibility for $\lambda x.t$ to evolve only if a value v is provided by the environment.

The rules for context capture are built following the principles of complementary semantics developed in [57]. The label of the transition $t \xrightarrow{E} t'$ contains what the environment needs to provide (a context E, but also an enclosing reset, left implicit) for the control stuck term t to reduce to t'. Hence, the transition $t \xrightarrow{E} t'$ means that we have $\langle E[t]\rangle \xrightarrow{\tau} t'$ by context capture. For example, in the rule (*shift*), the result of the capture of E by $\mathcal{S}k.t$ is $\langle t\{\lambda x.\langle E[x]\rangle/k\}\rangle$.

In rule $(\mathit{left}_{\mathcal{S}})$, we want to know the result of the capture of E by the term $t_0\ t_1$, assuming t_0 contains a shift ready to perform the capture. Under this hypothesis, the capture of E by $t_0\ t_1$ comes from the capture of $E[\Box\ t_1]$ by t_0. Therefore, as a premise of the rule $(\mathit{left}_{\mathcal{S}})$, we check that t_0 is able to capture $E[\Box\ t_1]$, and the result t_0' of this transition is exactly the result we want for the capture of E by $t_0\, t_1$. The rule ($right_{\mathcal{S}}$) follows the same pattern. Finally, a control stuck term t enclosed in a reset is able to perform an internal action (rule ($\langle\cdot\rangle_{\mathcal{S}}$)); we obtain the result t' of the transition $\langle t\rangle \xrightarrow{\tau} t'$ by letting t capture the empty context, i.e., by considering the transition $t \xrightarrow{\square} t'$.

Example 5.1. With the same notations as in Example 2.1, we illustrate how the LTS handles capture by considering the transition from $\langle (i\, \mathcal{S}k.\omega)\; (\omega\, \omega)\rangle$.

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\mathcal{S}k.\omega \xrightarrow{(i\,\square)\;(\omega\,\omega)} \langle\omega\rangle}\;(shift)}{i\;\mathcal{S}k.\omega \xrightarrow{\square\;(\omega\,\omega)} \langle\omega\rangle}\;(right_{\mathcal{S}})}{(i\;\mathcal{S}k.\omega)\;(\omega\,\omega) \xrightarrow{\square} \langle\omega\rangle}\;(left_{\mathcal{S}})}{\langle (i\;\mathcal{S}k.\omega)\;(\omega\,\omega)\rangle \xrightarrow{\tau} \langle\omega\rangle}\;(\langle\cdot\rangle_{\mathcal{S}})$$

Reading the tree from bottom to top, we see that the rules ($\langle\cdot\rangle_{\mathcal{S}}$), ($left_{\mathcal{S}}$), and ($right_{\mathcal{S}}$) build the captured context in the label by deconstructing the initial term. Indeed, the rule ($\langle\cdot\rangle_{\mathcal{S}}$) removes the outermost reset and initiates the context in the label with $\square$. The rules ($left_{\mathcal{S}}$) and ($right_{\mathcal{S}}$) then successively remove the outermost application and store it in the context. The process continues until a shift operator is found; then we know the captured context is completed, and the rule ($shift$) computes the result of the capture. This result is then simply propagated from top to bottom by the other rules.

We now prove that the LTS corresponds to the reduction semantics $\rightarrow_{\mathrm{v}}$ and exhibits the observable terms (values and control stuck terms) of the language. The only difficulty is in the treatment of control stuck terms. The next lemma makes the correspondence between $\xrightarrow{E}$ and control stuck terms explicit.

Lemma 5.2. *If $t \xrightarrow{E} t'$, then there exist E', k, and s such that $t = E'[\mathcal{S}k.s]$ and $t' = \langle s\{\lambda x.\langle E[E'[x]]\rangle/k\}\rangle$.*

The proof is direct by induction on $t \xrightarrow{E} t'$. From this lemma, we can deduce the correspondence between $\xrightarrow{\tau}$ and $\rightarrow_{\mathrm{v}}$, and between $\xrightarrow{\alpha}$ (for $\alpha \neq \tau$) and the observable actions of the language.

Proposition 5.3. *The following hold:*

- *We have $\xrightarrow{\tau} = \rightarrow_{\mathrm{v}}$.*
- *If $t \xrightarrow{E} t'$, then t is a stuck term, and $\langle E[t]\rangle \xrightarrow{\tau} t'$.*
- *If $t \xrightarrow{v} t'$, then t is a value, and $t\, v \xrightarrow{\tau} t'$.*

Proof. For the first result, the only difficult transition to check is the capture by shift. If $\langle t\rangle \xrightarrow{\tau} t'$ with $t \xrightarrow{\square} t'$, then by Lemma 5.2, there exists E, s such that $t = E[\mathcal{S}k.s]$ and $t' = \langle s\{\lambda x.\langle E[x]\rangle/k\}\rangle$; therefore, $\langle E[\mathcal{S}k.s]\rangle \rightarrow_{\mathrm{v}}$

$\langle s\{\lambda x.\langle E[x]\rangle/k\}\rangle$ holds, as wished. For the opposite direction, if $F[\langle E[\mathcal{S}k.s]\rangle] \rightarrow_{\mathrm{v}} F[\langle s\{\lambda x.\langle E[x]\rangle/k\}\rangle]$, then $F[\langle E[\mathcal{S}k.s]\rangle] \xrightarrow{\tau} F[\langle s\{\lambda x.\langle E[x]\rangle/k\}\rangle]$ holds by reasoning by induction on F.

The second and third results are straightforward, either by using Lemma 5.2, or by examining the LTS rules.

5.2. Applicative Bisimilarity

We now define the notion of applicative bisimilarity for $\lambda_{\mathcal{S}}$. We write $\Rightarrow$ for the reflexive and transitive closure of $\xrightarrow{\tau}$. We define the weak delay transition[2] $\stackrel{\alpha}{\Rightarrow}$ as $\Rightarrow$ if $\alpha = \tau$ and as $\Rightarrow\xrightarrow{\alpha}$ otherwise. The definition of (weak delay) bisimilarity is then straightforward.

Definition 5.4. A relation $\mathcal{R}$ on closed terms is an applicative simulation if $t_0 \mathrel{\mathcal{R}} t_1$ implies that for all $t_0 \xrightarrow{\alpha} t_0'$, there exists t_1' such that $t_1 \stackrel{\alpha}{\Rightarrow} t_1'$ and $t_0' \mathrel{\mathcal{R}} t_1'$. A relation $\mathcal{R}$ on closed terms is an applicative bisimulation if $\mathcal{R}$ and $\mathcal{R}^{-1}$ are applicative simulations. Applicative bisimilarity $\mathbb{A}$ is the largest applicative bisimulation.

In words, two terms are equivalent if any transition from one is matched by a weak transition with the same label from the other. As in the λ-calculus [1, 29], it is not mandatory to test the internal steps when proving that two terms are bisimilar, because of the following result.

Proposition 5.5. *If $t \xrightarrow{\tau} t'$ (respectively $t \Downarrow_{\mathrm{v}} t'$) then $t \mathrel{\mathbb{A}} t'$.*

Proposition 5.5 holds because $\{(t, t') \mid t \xrightarrow{\tau} t'\}$ is an applicative bisimulation. Consequently, applicative bisimulation can be defined in terms of big-step transitions as follows.

Definition 5.6. A relation $\mathcal{R}$ on closed terms is a big-step applicative simulation if $t_0 \mathrel{\mathcal{R}} t_1$ implies that for all $t_0 \stackrel{\alpha}{\Rightarrow} t_0'$ with $\alpha \neq \tau$, there exists t_1' such that $t_1 \stackrel{\alpha}{\Rightarrow} t_1'$ and $t_0' \mathrel{\mathcal{R}} t_1'$. A relation $\mathcal{R}$ on closed terms is a big-step applicative bisimulation if $\mathcal{R}$ and $\mathcal{R}^{-1}$ are big-step applicative simulations.

Proposition 5.7. *If $\mathcal{R}$ is a big-step applicative bisimulation, then $\mathcal{R} \subseteq \mathbb{A}$.*

In this section, we drop the adjective "applicative" and refer to the two kinds of relations simply as "bisimulation" and "big-step bisimulation" where it does not cause confusion. We work with both styles (small-step and big-step), depending on which one is easier to use in a given proof.

Example 5.8 (double reset). We show that $\langle\langle t\rangle\rangle \mathrel{\mathbb{A}^\circ} \langle t\rangle$ holds by proving that $\mathcal{R} \stackrel{\mathsf{def}}{=} \{(\langle t\rangle, \langle\langle t\rangle\rangle) \mid t \in \mathcal{T}_c\} \cup \{(t,t) \mid t \in \mathcal{T}_c\}$ is a big-step applicative bisimulation. If $\langle t\rangle$ and/or $\langle\langle t\rangle\rangle$ is open, then $\langle t\rangle\sigma = \langle t\sigma\rangle$ (and similarly with $\langle\langle t\rangle\rangle$), for all closing substitution σ, so we still have terms in $\mathcal{R}$. With closed terms, the only

[2] A transition where internal steps are allowed before, but not after a visible action.

possible (big-step) transition is $\langle t\rangle \overset{v}{\Rightarrow} t'$, which means $\langle t\rangle \Downarrow_{\mathrm{v}} v' \xrightarrow{v} t'$. But we have proved in Example 4.3 that $\langle t\rangle \Downarrow_{\mathrm{v}} v'$ iff $\langle\langle t\rangle\rangle \Downarrow_{\mathrm{v}} v'$. Consequently, we have $\langle t\rangle \overset{v}{\Rightarrow} t'$ iff $\langle\langle t\rangle\rangle \overset{v}{\Rightarrow} t'$, and we have $t' \mathcal{R} t'$, as wished. The structure of the proof is simpler than in Example 4.3 because we do not have to consider open stuck terms.

Example 5.9 (Turing's combinator). We now consider Turing's combinator Θ and its variant $\Theta_{\mathcal{S}} \overset{\mathsf{def}}{=} \langle\theta\, \mathcal{S}k.k\; k\rangle$. The two terms can perform the following transitions:

$$\Theta \overset{v}{\Rightarrow} v\;(\lambda z.\theta\;\theta\;v\;z)$$
$$\Theta_{\mathcal{S}} \overset{v}{\Rightarrow} v\;(\lambda z.(\lambda x.\langle\theta\;x\rangle)\;(\lambda x.\langle\theta\;x\rangle)\;v\;z)$$

Assuming $v = \lambda x.t$, we have to study the behavior of $t\{(\lambda z.\theta\;\theta\;v\;z)/x\}$, and $t\{(\lambda z.(\lambda x.\langle\theta\;x\rangle)\;(\lambda x.\langle\theta\;x\rangle)\;v\;z)/x\}$. A way to proceed is by case analysis on t, the interesting case being $t = F[x\;v']$. The resulting applicative bisimulation one can write to relate Θ and $\Theta_{\mathcal{S}}$ is much more complex than the normal-form bisimulation of Example 4.4.

Remark 5.10. Applicative simulation can be formulated in a more classic, but equivalent, way (without labeled transitions), as follows. A relation $\mathcal{R}$ on closed terms is an applicative simulation if $t_0 \mathcal{R} t_1$ implies:

- if $t_0 \rightarrow_{\mathrm{v}} t'_0$, then there exists t'_1 such that $t_1 \rightarrow^*_{\mathrm{v}} t'_1$ and $t'_0 \mathcal{R} t'_1$;
- if t_0 is a value $\lambda x.t'_0$, then there exists t'_1 such that $t_1 \rightarrow^*_{\mathrm{v}} \lambda x.t'_1$, and for all closed v, we have $t'_0\{v/x\} \mathcal{R} t'_1\{v/x\}$;
- if t_0 is a stuck term $E_0[\mathcal{S}k.t'_0]$, then there exist t'_1 and E_1 such that $t_1 \rightarrow^*_{\mathrm{v}} E_1[\mathcal{S}k.t'_1]$, and for all closed E, we have $\langle t'_0\{\lambda x.\langle E[E_0[x]]\rangle/k\}\rangle \mathcal{R} \langle t'_1\{\lambda x.\langle E[E_1[x]]\rangle/k\}\rangle$.

5.3. Soundness

To prove the soundness of $\mathbb{A}$ w.r.t. the contextual equivalence $\mathbb{C}$, we show that $\mathbb{A}$ is a congruence using *Howe's method*, a well-known congruence proof method initially developed for the λ-calculus [34, 29]. We briefly explain how we apply it to $\mathbb{A}$; the complete proof can be found in Appendix B. The idea of the method is as follows: first, define the *Howe's closure* of $\mathbb{A}$, written $\mathbb{A}^\bullet$, a relation which contains $\mathbb{A}$ and is compatible by construction. Then, prove a simulation-like property for $\mathbb{A}^\bullet$; from this result, prove that $\mathbb{A}^\bullet$ and $\mathbb{A}$ coincide on closed terms. Because $\mathbb{A}^\bullet$ is compatible, it shows that $\mathbb{A}$ is compatible as well, and therefore a congruence.

The definition of $\mathbb{A}^\bullet$ relies on the notion of *compatible refinement*; given a relation $\mathcal{R}$ on open terms, the compatible refinement $\widehat{\mathcal{R}}$ relates two terms iff they have the same outermost operator and their immediate subterms are related by $\mathcal{R}$. Formally, it is inductively defined by the following rules:

$$\frac{}{x \mathrel{\widehat{\mathcal{R}}} x} \qquad \frac{t_0 \mathrel{\mathcal{R}} t_1}{\lambda x.t_0 \mathrel{\widehat{\mathcal{R}}} \lambda x.t_1} \qquad \frac{t_0 \mathrel{\mathcal{R}} t_1 \qquad t_0' \mathrel{\mathcal{R}} t_1'}{t_0\, t_0' \mathrel{\widehat{\mathcal{R}}} t_1\, t_1'}$$

$$\frac{t_0 \mathrel{\mathcal{R}} t_1}{\mathcal{S}k.t_0 \mathrel{\widehat{\mathcal{R}}} \mathcal{S}k.t_1} \qquad \frac{t_0 \mathrel{\mathcal{R}} t_1}{\langle t_0 \rangle \mathrel{\widehat{\mathcal{R}}} \langle t_1 \rangle}$$

Howe's closure $\mathbb{A}^\bullet$ is inductively defined as the smallest compatible relation containing $\mathbb{A}^\circ$ and closed under right composition with $\mathbb{A}^\circ$.

Definition 5.11. Howe's closure $\mathbb{A}^\bullet$ is the smallest relation satisfying:

$$\frac{t_0 \mathrel{\mathbb{A}^\circ} t_1}{t_0 \mathrel{\mathbb{A}^\bullet} t_1} \qquad \frac{t_0 \mathrel{\mathbb{A}^\bullet\mathbb{A}^\circ} t_1}{t_0 \mathrel{\mathbb{A}^\bullet} t_1} \qquad \frac{t_0 \mathrel{\widehat{\mathbb{A}^\bullet}} t_1}{t_0 \mathrel{\mathbb{A}^\bullet} t_1}$$

By construction, $\mathbb{A}^\bullet$ is compatible (by the third rule of the definition), and composing on the right with $\mathbb{A}^\circ$ gives some transitivity properties to $\mathbb{A}^\bullet$. In particular, we can prove $\mathbb{A}^\bullet$ is *substitutive*: if $t_0 \mathrel{\mathbb{A}^\bullet} t_1$ and $v_0 \mathrel{\mathbb{A}^\bullet} v_1$, then $t_0\{v_0/x\} \mathrel{\mathbb{A}^\bullet} t_1\{v_1/x\}$.

Let $(\mathbb{A}^\bullet)^c$ be the restriction of $\mathbb{A}^\bullet$ to closed terms. We cannot prove directly that $(\mathbb{A}^\bullet)^c$ is a bisimulation, so we prove a stronger result. Suppose we have $t_0 \mathrel{(\mathbb{A}^\bullet)^c} t_1$; instead of simply requiring $t_0 \xrightarrow{\alpha} t_0'$ to be matched by t_1 with the same label α, we ask t_1 to be able to respond for any label α' related to α by $(\mathbb{A}^\bullet)^c$. We, therefore, extend $\mathbb{A}^\bullet$ to all labels, by adding the relation $\tau \mathrel{\mathbb{A}^\bullet} \tau$, and by defining $E \mathrel{\mathbb{A}^\bullet} E'$ as follows:

$$\frac{}{\Box \mathrel{\mathbb{A}^\bullet} \Box} \qquad \frac{E_0 \mathrel{\mathbb{A}^\bullet} E_1 \qquad t_0 \mathrel{\mathbb{A}^\bullet} t_1}{E_0\, t_0 \mathrel{\mathbb{A}^\bullet} E_1\, t_1} \qquad \frac{E_0 \mathrel{\mathbb{A}^\bullet} E_1 \qquad v_0 \mathrel{\mathbb{A}^\bullet} v_1}{v_0\, E_0 \mathrel{\mathbb{A}^\bullet} v_1\, E_1}$$

Lemma 5.12 (Simulation-like property). *If $t_0 \mathrel{(\mathbb{A}^\bullet)^c} t_1$ and $t_0 \xrightarrow{\alpha} t_0'$, then for all $\alpha \mathrel{(\mathbb{A}^\bullet)^c} \alpha'$, there exists t_1' such that $t_1 \xRightarrow{\alpha'} t_1'$ and $t_0' \mathrel{(\mathbb{A}^\bullet)^c} t_1'$.*

The main difficulty when applying Howe's method is to prove this simulation-like property. The proof (in Appendix B) is by induction on $t_0 \mathrel{(\mathbb{A}^\bullet)^c} t_1$, and then by case analysis on the transition $t_0 \xrightarrow{\alpha} t_0'$. Lemma 5.12 allows us to prove that $(\mathbb{A}^\bullet)^c$ is a simulation, by choosing $\alpha' = \alpha$. We cannot directly deduce that $(\mathbb{A}^\bullet)^c$ is a bisimulation, however we can prove that its transitive and reflexive closure $((\mathbb{A}^\bullet)^c)^*$ is a bisimulation, because of the following classical property of the Howe's closure (for the proof of this result, see, e.g., [29]).

Lemma 5.13. *The relation $(\mathbb{A}^\bullet)^*$ is symmetric.*

The fact that $((\mathbb{A}^\bullet)^c)^*$ is a bisimulation implies that $((\mathbb{A}^\bullet)^c)^* \subseteq \mathbb{A}$. Because $\mathbb{A} \subseteq (\mathbb{A}^\bullet)^c \subseteq ((\mathbb{A}^\bullet)^c)^*$ holds by construction, we can deduce $\mathbb{A} = (\mathbb{A}^\bullet)^c$. Since $(\mathbb{A}^\bullet)^c$ is compatible, and we can easily show that $\mathbb{A}$ is transitive and reflexive, we have the following result.

Theorem 5.14. *The relation $\mathbb{A}$ is a congruence.*

As a corollary, $\mathbb{A}$ is sound w.r.t. contextual equivalence.

Theorem 5.15. *We have $\mathbb{A} \subseteq \mathbb{C}$.*

Proof. Let $t_0 \mathbb{A} t_1$, and C be a closed context. Because $\mathbb{A}$ is a congruence, we have $C[t_0] \mathbb{A} C[t_1]$. If $C[t_0] \Downarrow_{\text{v}} v_0$, then we have $C[t_0] \overset{\tau}{\Rightarrow} v_0 \xrightarrow{v} t'_0$ for some t'_0; by bisimilarity, there exist v_1, t'_1 such that $C[t_1] \overset{\tau}{\Rightarrow} v_1 \xrightarrow{v} t'_1$. Therefore, we have $C[t_1] \Downarrow_{\text{v}} v_1$. The reasoning is the same if $C[t_0] \Downarrow_{\text{v}} t'_0$ with t'_0 control stuck, and for the evaluations of $C[t_1]$.

5.4. Completeness and Context Lemma

We now prove that $\mathbb{A}$ is complete w.r.t. $\mathbb{C}$, and deduce some results about $\mathbb{C}$ in the process, such as a context lemma showing that testing terms with evaluation contexts is as discriminating as testing with any contexts. We remind that we write $\mathbb{D}$ for the contextual equivalence which tests with contexts F only (see Definition 3.7). We already have $\mathbb{C} \subseteq \mathbb{D}$ by definition. We first prove that $\mathbb{A}$ is complete w.r.t. $\mathbb{D}$.

Theorem 5.16. *We have $\mathbb{D} \subseteq \mathbb{A}$.*

Proof. Because $\mathbb{D}$ is symmetric, it is enough to prove that $\mathbb{D}$ is a big-step simulation. Let $t_0 \mathbb{D} t_1$. We have two cases to consider.

Suppose $t_0 \overset{v}{\Rightarrow} t'_0$. Then we have $t_0 \Downarrow_{\text{v}} v_0$ for some v_0. By definition of $\mathbb{D}$, there exists v_1 such that $t_1 \Downarrow_{\text{v}} v_1$. Therefore, we have $t_1 \overset{v}{\Rightarrow} t'_1$ for some t'_1 and $t_0\, v \overset{\tau}{\Rightarrow} t'_0$ and $t_1\, v \overset{\tau}{\Rightarrow} t'_1$ by Proposition 5.3. Hence, we have $t_0\, v \mathbb{A} t'_0$ and $t_1\, v \mathbb{A} t'_1$ by Proposition 5.5. Because $t_0 \mathbb{D} t_1$, we have $t_0\, v \mathbb{D} t_1\, v$. Finally, we have $t'_0 \mathbb{A} t_0\, v \mathbb{D} t_1\, v \mathbb{A} t'_1$ and $t'_0 \mathbb{D} t'_1$ by Theorem 5.15 and transitivity of $\mathbb{D}$.

Suppose $t_0 \overset{E}{\Rightarrow} t'_0$. Then we have $t_0 \Downarrow_{\text{v}} t''_0$ for some t''_0. By definition of $\mathbb{D}$, there exists t''_1 such that $t_1 \Downarrow_{\text{v}} t''_1$. Therefore, by definition of the LTS, we have $t_1 \overset{E}{\Rightarrow} t'_1$ for some t'_1. We then have $\langle E[t_0]\rangle \overset{\tau}{\Rightarrow} t'_0$ and $\langle E[t_1]\rangle \overset{\tau}{\Rightarrow} t'_1$ by Proposition 5.3. Hence, we have $\langle E[t_0]\rangle \mathbb{A} t'_0$ and $\langle E[t_1]\rangle \mathbb{A} t'_1$ by Proposition 5.5. Because $t_0 \mathbb{D} t_1$, we have $\langle E[t_0]\rangle \mathbb{D} \langle E[t_1]\rangle$. Finally, we have $t'_0 \mathbb{A} \langle E[t_0]\rangle \mathbb{D} \langle E[t_1]\rangle \mathbb{A} t'_1$ and, therefore, we have $t'_0 \mathbb{D} t'_1$ by Theorem 5.15 and transitivity of $\mathbb{D}$.

As a result, the relations $\mathbb{C}$, $\mathbb{D}$, and $\mathbb{A}$ coincide, which means that $\mathbb{A}$ is complete w.r.t. $\mathbb{C}$

Corollary 5.17. *We have $\mathbb{C} = \mathbb{D} = \mathbb{A}$.*

Indeed, we have $\mathbb{D} \subseteq \mathbb{A}$ (Theorem 5.16), $\mathbb{A} \subseteq \mathbb{C}$ (Theorem 5.15), and $\mathbb{C} \subseteq \mathbb{D}$ (by definition).

This equality also allows us to prove that we can formulate the open extension of $\mathbb{C}$ using capturing contexts.

Proposition 5.18. *We have $t_0 \mathbb{C}^\circ t_1$ iff for all C capturing the variables of t_0 and t_1, the following holds:*

- $C[t_0] \Downarrow_{\mathrm{v}} v_0$ *iff* $C[t_1] \Downarrow_{\mathrm{v}} v_1$*;*
- $C[t_0] \Downarrow_{\mathrm{v}} t'_0$*, where t'_0 is control stuck, iff $C[t_1] \Downarrow_{\mathrm{v}} t'_1$, with t'_1 control stuck as well.*

Proof. Suppose $t_0 \mathbb{C}^\circ t_1$. Then $t_0 \mathbb{A}^\circ t_1$, and because $\mathbb{A}^\circ$ is a congruence, for all C capturing the variables of t_0 and t_1, we have $C[t_0] \mathbb{A} C[t_1]$. We have $C[t_0] \Downarrow_{\mathrm{v}} v_0$ iff $C[t_1] \Downarrow_{\mathrm{v}} v_1$ by bisimilarity definition, and similarly with $C[t_0] \Downarrow_{\mathrm{v}} t'_0$, where t'_0 is control stuck.

For the reverse implication, suppose that for all C capturing the variables of t_0 and t_1, the two items of the proposition hold. Let $\sigma = \{v_1/x_1 \dots v_n/x_n\}$ be a substitution closing t_0 and t_1. Let C be a closed context. We want to prove that $C[t_0\sigma] \rightarrow_{\mathrm{v}}^* v$ iff $C[t_1\sigma] \rightarrow_{\mathrm{v}}^* v'$ for some v and v', and similarly for control stuck terms. But the context $C' \stackrel{\mathsf{def}}{=} C[(\lambda x_1 \dots x_n.\Box)\, v_1\ \dots v_n]$ is a context capturing the variables of t_0 and t_1, and we have $C'[t_0] \rightarrow_{\mathrm{v}}^* C[t_0\sigma]$ and $C'[t_1] \rightarrow_{\mathrm{v}}^* C[t_1\sigma]$. Consequently, $C[t_0\sigma] \rightarrow_{\mathrm{v}}^* v$ iff $C'[t_0] \rightarrow_{\mathrm{v}}^* v$ iff $C'[t_0] \rightarrow_{\mathrm{v}}^* v'$ (first item of the proposition) iff $C[t_1\sigma] \rightarrow_{\mathrm{v}}^* v'$ for some v and v'. The reasoning is the same for control stuck terms.

The next example completes the proof of Proposition 4.7: we show that the terms which cannot be equated with normal-form bisimulation are applicative bisimilar and, therefore, contextually equivalent.

Proposition 5.19. *Let $i \stackrel{\mathsf{def}}{=} \lambda x.x$; then $\langle\langle x\, i\rangle\, \mathcal{S}k.i\rangle \mathbb{C}^\circ \langle\langle x\, i\rangle\, (\langle x\, i\rangle\, \mathcal{S}k.i)\rangle$.*

Proof. We prove that $\mathcal{R} \stackrel{\mathsf{def}}{=} \{(\langle\langle t\rangle\, \mathcal{S}k.i\rangle, \langle\langle t\rangle\, (\langle t\rangle\, \mathcal{S}k.i)\rangle) \mid t \in \mathcal{T}_c\} \cup \{(t,t) \mid t \in \mathcal{T}_c\}$ is a big-step bisimulation. The term $\langle t\rangle$ can either diverge or reduce to a value (according to Proposition 2.9). If it diverges, then both $\langle\langle t\rangle\, \mathcal{S}k.i\rangle$ and $\langle\langle t\rangle\, (\langle t\rangle\, \mathcal{S}k.i)\rangle$ diverge, otherwise, they both evaluate to i. For all v', we, therefore, have $\langle\langle t\rangle\, \mathcal{S}k.i\rangle \stackrel{v'}{\Rightarrow} v'$ and $\langle\langle t\rangle\, (\langle t\rangle\, \mathcal{S}k.i)\rangle \stackrel{v'}{\Rightarrow} v'$, and $v' \mathcal{R} v'$ holds, as wished.

5.5. Proving the Axioms

As with normal-form bisimulation (Section 4.5), we show how to prove Kameyama and Hasegawa's axioms (Section 2.4) except for $\mathcal{S}_{\mathtt{elim}}$ using applicative bisimulation. In the following propositions, we assume the terms to be closed, since the proofs for open terms can be deduced directly from the results for closed terms. First, note that the β_{v}, $\langle\cdot\rangle_{\mathcal{S}}$, and $\langle\cdot\rangle_{\mathtt{val}}$ axioms are direct consequences of Proposition 5.5.

Proposition 5.20 (η_{v} axiom). *If $x \notin \mathsf{fv}(v)$, then $\lambda x.v\, x \mathbb{A} v$.*

Proof. We prove that $\mathcal{R} \stackrel{\mathsf{def}}{=} \{(\lambda x.(\lambda y.t)\, x, \lambda y.t) \mid t \in \mathcal{T}, \mathsf{fv}(t) \subseteq \{y\}\} \cup \mathbb{A}$ is a bisimulation. To this end, we have to check that $\lambda x.(\lambda y.t)\, x \xrightarrow{v_0} (\lambda y.t)\, v_0$ is matched by $\lambda y.t \xrightarrow{v_0} t\{v_0/y\}$, i.e., that $(\lambda y.t)\, v_0 \;\mathcal{R}\; t\{v_0/y\}$ holds for all v_0. We have $(\lambda y.t)\, v_0 \xrightarrow{\tau} t\{v_0/y\}$, and because $\xrightarrow{\tau} \subseteq \mathbb{A} \subseteq \mathcal{R}$, we have the required result.

Proposition 5.21 ($\mathcal{S}_{\langle\cdot\rangle}$ axiom). *We have $\mathcal{S}k.\langle t\rangle \;\mathbb{A}\; \mathcal{S}k.t$.*

Proof. We have $\mathcal{S}k.\langle t\rangle \xrightarrow{E} \langle\langle t\{\lambda x.\langle E[x]\rangle/k\}\rangle\rangle$ and $\mathcal{S}k.t \xrightarrow{E} \langle t\{\lambda x.\langle E[x]\rangle/k\}\rangle$ for all E. We obtain terms of the form $\langle\langle t'\rangle\rangle$ and $\langle t'\rangle$, and we have proved in Example 5.8 that $\langle\langle t'\rangle\rangle \;\mathbb{A}\; \langle t'\rangle$ holds for all t'.

Proposition 5.22 ($\langle\cdot\rangle_{\mathtt{lift}}$ axiom). *We have $\langle(\lambda x.t_0)\, \langle t_1\rangle\rangle \;\mathbb{A}\; (\lambda x.\langle t_0\rangle)\, \langle t_1\rangle$.*

Proof. A transition $\langle(\lambda x.t_0)\, \langle t_1\rangle\rangle \stackrel{\alpha}{\Rightarrow} t'$ (with $\alpha \neq \tau$) is possible only if $\langle t_1\rangle$ evaluates to some value v (evaluation to a control stuck terms is not possible according to Proposition 2.9). In this case, we have $\langle(\lambda x.t_0)\, \langle t_1\rangle\rangle \stackrel{\tau}{\Rightarrow} \langle(\lambda x.t_0)\, v\rangle \xrightarrow{\tau} \langle t_0\{v/x\}\rangle$ and $(\lambda x.\langle t_0\rangle)\, \langle t_1\rangle \stackrel{\tau}{\Rightarrow} \langle t_0\{v/x\}\rangle$. Therefore, we have $\langle(\lambda x.t_0)\, \langle t_1\rangle\rangle \stackrel{\alpha}{\Rightarrow} t'$ (with $\alpha \neq \tau$) iff $(\lambda x.\langle t_0\rangle)\, \langle t_1\rangle \stackrel{\alpha}{\Rightarrow} t'$. From there, it is easy to conclude.

Proposition 5.23 (β_Ω axiom). *If $x \notin \mathsf{fv}(E)$, then $(\lambda x.E[x])\, t \;\mathbb{A}\; E[t]$.*

Proof (Sketch). We first give some intuitions on why the proof of this result is harder with applicative bisimulation than with normal-form bisimulation. The difficult case is when t in the initial terms $(\lambda x.E[x])\, t$ and $E[t]$ is a control stuck term $E_0[\mathcal{S}k.t']$. Then we have the following transitions:

$$(\lambda x.E[x])\, t \xrightarrow{E_1} \langle t'\{\lambda y.\langle E_1[(\lambda x.E[x])\, E_0[y]]\rangle/k\}\rangle$$
$$E[t] \xrightarrow{E_1} \langle t'\{\lambda y.\langle E_1[E[E_0[y]]]\rangle/k\}\rangle$$

We obtain terms of the form $\langle t'\rangle\sigma$ and $\langle t'\rangle\sigma'$ (where σ and σ' are the above substitutions). We now have to consider the transitions from these terms, and the interesting case is when $\langle t'\rangle = F[k\, v]$.

$$\langle t'\rangle\sigma \xrightarrow{\tau} F\sigma[\langle E_1[(\lambda x.E[x])\, E_0[v\sigma]]\rangle] \stackrel{\mathsf{def}}{=} t_0$$
$$\langle t'\rangle\sigma' \xrightarrow{\tau} F\sigma'[\langle E_1[E[E_0[v\sigma']]]\rangle] \stackrel{\mathsf{def}}{=} t_1$$

We obtain terms that are similar to the initial terms $(\lambda x.E[x])t$ and $E[t]$, except for the extra contexts F and E_1, and the substitutions σ and σ'. Again, the interesting cases are when $E_0[v]$ is either a control stuck term, or a term of the form $F'[k\, v']$. Looking at these cases, we see that the bisimulation we have to define has to relate terms similar to t_0 and t_1, except with an arbitrary number of contexts F' and substitutions similar to σ and σ'.

Formally, given a sequence of (continuation) variables $k_1, \ldots, k_n$ and a sequence $\overrightarrow{E_k}$ of triples of contexts E_i, E'_i, E''_i such that

$$\mathsf{fv}(E_i) \cup \mathsf{fv}(E'_i) \cup \mathsf{fv}(E''_i) \subseteq \{k_1, \ldots, k_{i-1}\} \qquad \text{for } 1 \leq i \leq n \qquad (\star)$$

we define two families of sequences of substitutions as follows:

$$\begin{array}{rcl}\sigma_i^{\overrightarrow{E}} & = & \{\lambda x.\langle E''_i[(\lambda y.E_i[y])\ E'_i[x]]\rangle/k_i\} \\ \delta_i^{\overrightarrow{E}} & = & \{\lambda x.\langle E''_i[E_i[E'_i[x]]]\rangle/k_i\}\end{array}$$

Additionally, given a term t, a sequence of pure contexts $\overrightarrow{E} = E_1, \ldots, E_m$ and a sequence of evaluation contexts $\overrightarrow{F} = F_1, \ldots, F_m$, we inductively define two sequences of terms, $s_0, \ldots, s_m$ and $u_0, \ldots, u_m$, as follows:

$$\begin{array}{rclrcl}s_0^{t,\overrightarrow{E},\overrightarrow{F}} & = & t & u_0^{t,\overrightarrow{E},\overrightarrow{F}} & = & t \\ s_{i+1}^{t,\overrightarrow{E},\overrightarrow{F}} & = & F_{i+1}[(\lambda x.E_{i+1}[x])\ s_i] & u_{i+1}^{t,\overrightarrow{E},\overrightarrow{F}} & = & F_{i+1}[E_{i+1}[u_i]]\end{array}$$

Then the following relation $\mathcal{R}$ is a bisimulation:

$$\begin{array}{rcl}\mathcal{R} & = & \{(s_i^{t,\overrightarrow{E},\overrightarrow{F}}\sigma_n^{\overrightarrow{E_k}} \ldots \sigma_1^{\overrightarrow{E_k}}, u_i^{t,\overrightarrow{E},\overrightarrow{F}}\delta_n^{\overrightarrow{E_k}} \ldots \delta_1^{\overrightarrow{E_k}}) \mid \\ & & \qquad \overrightarrow{k} = k_1, \ldots, k_n, n \geq 0, \\ & & \qquad \overrightarrow{E_k} \text{ satisfies } (\star), \\ & & \qquad \overrightarrow{E} = E_1, \ldots, E_m, \overrightarrow{F} = F_1, \ldots, F_m, m \geq 0, \\ & & \qquad \mathsf{fv}(t) \cup \mathsf{fv}(\overrightarrow{E}) \cup \mathsf{fv}(\overrightarrow{F}) \subseteq \{k_1, \ldots, k_n\}, \\ & & \qquad 0 \leq i \leq m\}\end{array}$$

We omit the complete bisimulation proof, as it is similar to the one in Appendix C.3: we proceed by case analysis on the transitions the related terms can perform.

5.6. Conclusion

In this section, we define an applicative bisimilarity for the relaxed semantics of $\lambda_{\mathcal{S}}$. As in the plain λ-calculus, our definition compares λ-abstractions by passing them a random value as argument. For control stuck terms, we run them in an arbitrary context E surrounded by a reset. Unlike normal-form bisimilarity, applicative bisimilarity is complete w.r.t. contextual equivalence, and we can easily prove equivalence of the terms in Proposition 4.7 with an applicative bisimulation.

Equivalence proofs can be simpler with applicative bisimulation than with normal-form bisimulation, because in the applicative case, we reason mainly on closed terms. As a result, a term cannot evaluate to an open stuck term, and we do not have to deal with this case, making the proofs shorter than with normal-form bisimulation (compare, e.g., Example 5.8 with Example 4.3, or Proposition 5.22 with Proposition 4.30). However, in general, the proofs are much simpler with normal-form bisimulation, because there is no quantification over arguments (values or contexts) in the λ-abstraction and control stuck terms cases. We can see the difference when comparing the proofs for fixed-point combinators (Example 4.4 and Example 5.9) and for the β_Ω axiom (Proposition 4.31 and Proposition 5.23). The lack of powerful up-to techniques such as bisimulation up to context makes also applicative bisimulation harder to use than the normal-form one.

6. Environmental Bisimilarity

Like applicative bisimilarity, environmental bisimilarity reduces closed terms to normal forms, which are then compared using some particular contexts (e.g., λ-abstractions are tested by passing them arguments). However, the testing contexts are not arbitrary, but built from an environment, which represents the knowledge acquired so far by an outside observer. The idea originally comes from languages with strict isolation or data abstraction [46, 85, 45, 84], where environments are used to handle information hiding. The term "environmental bisimulation" has then been introduced in [73, 74], and such a bisimilarity has been since defined in various higher-order languages (see, e.g., [76, 83, 66]). Environmental bisimilarity usually characterizes contextual equivalence, but is harder to use than applicative bisimilarity to prove that two given terms are equivalent. Nonetheless, one can define powerful up-to techniques [74] to simplify the equivalence proofs and deal with this extra difficulty. Besides, the authors of [44] argue that the additional complexity is necessary to handle more realistic features, like local state or exceptions. In addition, environmental bisimilarity does not require a particular proof technique, such as Howe's method, to prove its soundness; this will be helpful when considering the original semantics.

In what follows, we first give the definition of environmental bisimilarity for the relaxed semantics. We then discuss up-to techniques, and explain why they are not as helpful as, e.g., in the plain λ-calculus [74]. Finally, we propose a definition of environmental bisimilarity which we can prove complete for the original semantics. In this section, we focus on closed terms, before extending the relations we define to open terms.

6.1. Definition for the Relaxed Semantics

Environmental bisimulations use an environment $\mathcal{E}$ to accumulate knowledge about two tested terms. For the λ-calculus [74], $\mathcal{E}$ records the values (v_0, v_1) the tested terms reduce to, if they exist. We can then compare v_0 and v_1 at any time by passing them arguments built from $\mathcal{E}$. Even though in the relaxed semantics of $\lambda_{\mathcal{S}}$ control stuck terms are also normal forms, we do not store them in environments (like values), but we instead compare them during the bisimulation game, still using the environment. The rationale behind this change with our previous work [13], is explained in Remark 6.2.

As a result, an environment $\mathcal{E}$ is a relation on values; e.g., we define the identity environment $\mathcal{I}$ as $\{(v, v) \mid v \in \mathcal{T}_c\}$. We build terms and evaluation contexts from an environment $\mathcal{E}$ with the help of two closures, defined in Figure 7. Given a relation $\mathcal{R}$ on terms, we write $\dot{\mathcal{R}}$ for the term generating closure and $\ddot{\mathcal{R}}$ for the context generating closure. Even if $\mathcal{R}$ is defined only on closed terms, $\dot{\mathcal{R}}$ and $\ddot{\mathcal{R}}$ are defined on open terms and open contexts, respectively. In this section, we consider the restrictions of $\dot{\mathcal{R}}$ and $\ddot{\mathcal{R}}$ to respectively closed terms and closed contexts unless stated otherwise.

An environmental relation $\mathcal{X}$ is a set of environments $\mathcal{E}$, and triples $(\mathcal{E}, t_0, t_1)$, where t_0 and t_1 are closed. We write $t_0 \; \mathcal{X}_{\mathcal{E}} \; t_1$ as a shorthand for $(\mathcal{E}, t_0, t_1) \in \mathcal{X}$;

Term generating closure.

$$\frac{t \mathrel{\mathcal{R}} t'}{t \mathrel{\dot{\mathcal{R}}} t'} \qquad \frac{}{x \mathrel{\dot{\mathcal{R}}} x} \qquad \frac{t \mathrel{\dot{\mathcal{R}}} t'}{\lambda x.t \mathrel{\dot{\mathcal{R}}} \lambda x.t'} \qquad \frac{t_0 \mathrel{\dot{\mathcal{R}}} t_0' \qquad t_1 \mathrel{\dot{\mathcal{R}}} t_1'}{t_0\, t_1 \mathrel{\dot{\mathcal{R}}} t_0'\, t_1'} \qquad \frac{t \mathrel{\dot{\mathcal{R}}} t'}{\mathcal{S}k.t \mathrel{\dot{\mathcal{R}}} \mathcal{S}k.t'} \qquad \frac{t \mathrel{\dot{\mathcal{R}}} t'}{\langle t \rangle \mathrel{\dot{\mathcal{R}}} \langle t' \rangle}$$

Context generating closure.

$$\frac{}{\square \mathrel{\ddot{\mathcal{R}}} \square} \qquad \frac{F_0 \mathrel{\ddot{\mathcal{R}}} F_1 \qquad v_0 \mathrel{\dot{\mathcal{R}}} v_1}{v_0\, F_0 \mathrel{\ddot{\mathcal{R}}} v_1\, F_1} \qquad \frac{F_0 \mathrel{\ddot{\mathcal{R}}} F_1 \qquad t_0 \mathrel{\dot{\mathcal{R}}} t_1}{F_0\, t_0 \mathrel{\ddot{\mathcal{R}}} F_1\, t_1} \qquad \frac{F_0 \mathrel{\ddot{\mathcal{R}}} F_1}{\langle F_0 \rangle \mathrel{\ddot{\mathcal{R}}} \langle F_1 \rangle}$$

Figure 7: Term and context generating closures

roughly, it means that we test t_0 and t_1 with the knowledge $\mathcal{E}$. We define environmental bisimulation as follows.

Definition 6.1. A relation $\mathcal{X}$ is an environmental bisimulation if

1. $t_0 \mathrel{\mathcal{X}_\mathcal{E}} t_1$ implies:
   (a) if $t_0 \rightarrow_{\mathrm{v}} t_0'$, then there exists t_1' such that $t_1 \rightarrow_{\mathrm{v}}^* t_1'$ and $t_0' \mathrel{\mathcal{X}_\mathcal{E}} t_1'$;
   (b) if t_0 is a control stuck term $E_0[\mathcal{S}k.t_0']$, then there exists $E_1[\mathcal{S}k.t_1']$ such that $t_1 \Downarrow_{\mathrm{v}} E_1[\mathcal{S}k.t_1']$, and for all $E_0' \mathrel{\ddot{\mathcal{E}}} E_1'$, $\langle t_0'\{\lambda x.\langle E_0'[E_0[x]]\rangle/k\}\rangle \mathrel{\mathcal{X}_\mathcal{E}} \langle t_1'\{\lambda x.\langle E_1'[E_1[x]]\rangle/k\}\rangle$ holds for a fresh x;
   (c) if t_0 is a value, then there exists v_1 such that $t_1 \Downarrow_{\mathrm{v}} v_1$ and $\mathcal{E} \cup \{(t_0, v_1)\} \in \mathcal{X}$;
   (d) the converse of the above conditions on t_1;
2. for all $\mathcal{E} \in \mathcal{X}$, if $\lambda x.t_0 \mathrel{\mathcal{E}} \lambda x.t_1$ and $v_0 \mathrel{\dot{\mathcal{E}}} v_1$, then $t_0\{v_0/x\} \mathrel{\mathcal{X}_\mathcal{E}} t_1\{v_1/x\}$.

Environmental bisimilarity, written $\approx$, is the largest environmental bisimulation. To prove that two terms t_0 and t_1 are equivalent, we want to relate them without any predefined knowledge, i.e., we want to prove that $t_0 \approx_\emptyset t_1$ holds; we also write $\mathbb{E}$ for $\approx_\emptyset$. The relation $\mathbb{E}$ will be the candidate to characterize contextual equivalence.

The first part of the definition makes the bisimulation game explicit for t_0 and t_1, while the second part focuses on environments $\mathcal{E}$. If t_0 is a value, then t_1 has to evaluate to a value, and we extend the environment with the newly acquired knowledge. We then compare values in $\mathcal{E}$ (clause (2)) by applying them to arguments built from $\mathcal{E}$, as in the λ-calculus [74]. If t_0 is a control stuck term, then t_1 has to evaluate to a control stuck term as well, and we trigger the capture by putting them within contexts $\langle E_0' \rangle$, $\langle E_1' \rangle$ built from $\mathcal{E}$ (clause (1b)). This is similar to the way we test values and stuck terms with applicative bisimilarity (Section 5), except that applicative bisimilarity tests both values or stuck terms with the same argument or context. Using different entities (as in Definition 6.1) makes bisimulation proofs harder, but it simplifies the proof of congruence of the environmental bisimilarity.

Remark 6.2. The definition of environmental bisimulation in [13] treats control stuck terms the same way it deals with values, by storing them in an environment before testing them. As explained in [44], accumulating values in an environment to apply them to different arguments is justified (and, for some languages, necessary), because a value can be saved by a testing context and applied several times, like, e.g., in the context $(\lambda x.(x\, v_0)\,(x\, v_1))\,\Box$. Such a behavior is not possible with a control stuck term, since stuck terms are not values and cannot be saved. A testing context can, therefore, trigger the capture of a control stuck term only once, and not several times in different evaluation contexts. Consequently, we believe that control stuck terms should not be saved in environments.

Definition 6.1 is also easier to use than the definition of [13], because environments now contain only values, and control stuck terms are tested only once. For example, we can prove the β_Ω axiom with Definition 6.1 (and some basic up-to techniques; see Proposition 6.26), while we do not know how to prove it with the definition of [13] (even though we know it is possible, since the environmental bisimilarity of [13] is complete).

Unlike applicative bisimilarity or contextual equivalence, environmental bisimilarity tests terms with different arguments (built from the environment); we have to take this into account when extending environmental relations to open terms. If $\overrightarrow{x} = \mathsf{fv}(t_0) \cup \mathsf{fv}(t_1)$, then we write $t_0\ \mathcal{X}_{\mathcal{E}}{}^{\circ}\ t_1$ if $\lambda\overrightarrow{x}.t_0\ \mathcal{X}_{\mathcal{E}}\ \lambda\overrightarrow{x}.t_1$ holds. Note that to relate $\lambda\overrightarrow{x}.t_0$ and $\lambda\overrightarrow{x}.t_1$, Definition 6.1 replaces each variable with distinct closed values built from $\mathcal{E}$, while Definition 3.5, replaces each variable in both terms with the same arbitrary closed value.

As with the other styles of bisimilarity, we define a big-step variant of Definition 6.1, where we reason on the results of the evaluation directly.

Definition 6.3. A relation $\mathcal{X}$ is a big-step environmental bisimulation if

1. $t_0\ \mathcal{X}_{\mathcal{E}}\ t_1$ implies:
   (a) if $t_0 \Downarrow_{\mathrm{v}} v_0$, then there is v_1 such that $t_1 \Downarrow_{\mathrm{v}} v_1$ and $\mathcal{E} \cup \{(v_0, v_1)\} \in \mathcal{X}$;
   (b) if $t_0 \Downarrow_{\mathrm{v}} E_0[\mathcal{S}k.t_0']$, then there is $E_1[\mathcal{S}k.t_1']$ such that $t_1 \Downarrow_{\mathrm{v}} E_1[\mathcal{S}k.t_1']$, and $\langle t_0'\{\lambda x.\langle E_0'[E_0[x]]\rangle/k\}\rangle\ \mathcal{X}_{\mathcal{E}}\ \langle t_1'\{\lambda x.\langle E_1'[E_1[x]]\rangle/k\}\rangle$ holds for all $E_0'\ \ddot{\mathcal{E}}\ E_1'$ and a fresh x;
   (c) the converse of the above conditions on t_1;
2. for all $\mathcal{E} \in \mathcal{X}$, if $\lambda x.t_0\ \mathcal{E}\ \lambda x.t_1$ and $v_0\ \dot{\mathcal{E}}\ v_1$, then $t_0\{v_0/x\}\ \mathcal{X}_{\mathcal{E}}\ t_1\{v_1/x\}$.

Proposition 6.4. *If $\mathcal{X}$ is a big-step environmental bisimulation, then $\mathcal{X} \subseteq \approx$.*

We use the following results in the rest of the paper.

Proposition 6.5 (Weakening). *If $t_0 \approx_{\mathcal{E}} t_1$ and $\mathcal{E}' \subseteq \mathcal{E}$ then $t_0 \approx_{\mathcal{E}'} t_1$.*

A smaller environment is a weaker constraint, because we can build fewer arguments and contexts to test the normal forms in $\mathcal{E}$. The proof is as in [74]. Proposition 6.6 states that reduction (and, therefore, evaluation) is included in $\mathbb{E}$.

Proposition 6.6. *If* $t_0 \rightarrow_{\mathrm{v}} t'_0$, *then* $t_0 \mathbb{E} t'_0$.

We now give some examples showing how the notion of environmental bisimulation can be used.

Example 6.7 (double reset). We have $\langle\langle t\rangle\rangle \mathbb{E} \langle t\rangle$, because for a given t the relation

$$\{(\emptyset, \langle\langle t\rangle\rangle, \langle t\rangle)\} \cup \{(\mathcal{E}, t', t') \mid \mathcal{E} \subseteq \mathcal{I}, t' \in \mathcal{T}_c\} \cup \{\mathcal{E} \mid \mathcal{E} \subseteq \mathcal{I}\}$$

is a big-step environmental bisimulation. Indeed, we know that $\langle\langle t\rangle\rangle \Downarrow_{\mathrm{v}} v$ iff $\langle t\rangle \Downarrow_{\mathrm{v}} v$, so we have to consider environments $\mathcal{E}$ built out of pairs of values of the form (v, v). Then, testing such $\mathcal{E}$, suppose we take $\lambda x.t \; \mathcal{E} \; \lambda x.t$ and some arguments $v_0 \; \dot{\mathcal{E}} \; v_1$, and we need to relate $t\{v_0/x\}$ with $t\{v_1/x\}$. Since the terms related by $\mathcal{E}$ are the same, we have in fact $v_0 = v_1$, so we have to relate $t\{v_0/x\}$ with itself, hence the second set in the definition of the bisimulation.

Example 6.8 (Turing's combinator). Proving that Turing's combinator Θ is bisimilar to its variant $\Theta_{\mathcal{S}} \stackrel{\mathsf{def}}{=} \langle\theta \; \mathcal{S}k.k \; k\rangle$ using the basic definition of environmental bisimulation is harder than with applicative bisimulation (Example 5.9). We remind that

$$\Theta \Downarrow_{\mathrm{v}} \lambda y.y \; (\lambda z.\theta \; \theta \; y \; z) \stackrel{\mathsf{def}}{=} v_0, \text{ and}$$
$$\Theta_{\mathcal{S}} \Downarrow_{\mathrm{v}} \lambda y.y \; (\lambda z.(\lambda x.\langle\theta \; x\rangle) \; (\lambda x.\langle\theta \; x\rangle) \; y \; z) \stackrel{\mathsf{def}}{=} v_1.$$

Therefore, we have to put (v_0, v_1) in an environment $\mathcal{E}$. When we then test v_0 and v_1, we use arguments v'_0 and v'_1 such that $v'_0 \; \dot{\mathcal{E}} \; v'_1$, and we compare $v'_0 \; (\lambda z.\theta \; \theta \; v'_0 \; z)$ with $v'_1 \; (\lambda z.(\lambda x.\langle\theta \; x\rangle) \; (\lambda x.\langle\theta \; x\rangle) \; v'_1 \; z)$. Because we have two different terms v'_0 and v'_1, we can no longer do a case analysis as suggested in Example 5.9. To conclude with environmental bisimulation, we need bisimulation up-to context (see Example 6.22).

As usual with environmental relations, the candidate relation in Example 6.7 could be made simpler with the help of basic up-to techniques. In bisimulation up-to environment, one can use bigger environments than needed by Definition 6.1. As a result, instead of making the environment grow at each bisimulation step, we can directly use the largest possible environment.

Definition 6.9. A relation $\mathcal{X}$ is an environmental bisimulation up to environment if

1. $t_0 \; \mathcal{X}_{\mathcal{E}} \; t_1$ implies:
   (a) if $t_0 \rightarrow_{\mathrm{v}} t'_0$, then there exist t'_1 and $\mathcal{E}' \in \mathcal{X}$ such that $t_1 \rightarrow^*_{\mathrm{v}} t'_1$, $t'_0 \; \mathcal{X}_{\mathcal{E}'} \; t'_1$, and $\mathcal{E} \subseteq \mathcal{E}'$;
   (b) if t_0 is a stuck term $E_0[\mathcal{S}k.t'_0]$, then there exists $E_1[\mathcal{S}k.t'_1]$ such that $t_1 \Downarrow_{\mathrm{v}} E_1[\mathcal{S}k.t'_1]$, and for all $E'_0 \; \ddot{\mathcal{E}} \; E'_1$, there exists $\mathcal{E}'$ such that $\mathcal{E} \subseteq \mathcal{E}'$ and $\langle t'_0\{\lambda x.\langle E'_0[E_0[x]]\rangle/k\}\rangle \; \mathcal{X}_{\mathcal{E}'} \; \langle t'_1\{\lambda x.\langle E'_1[E_1[x]]\rangle/k\}\rangle$ for a fresh x;

   (c) if t_0 is a value, then there exist v_1 and $\mathcal{E}' \in \mathcal{X}$ such that $t_1 \Downarrow_{\text{v}} v_1$ and $\mathcal{E} \cup \{(t_0, t_1')\} \subseteq \mathcal{E}'$;
   (d) the converse of the above conditions on t_1;
2. for all $\mathcal{E} \in \mathcal{X}$, if $\lambda x.t_0 \; \mathcal{E} \; \lambda x.t_1$ and $v_0 \; \dot{\mathcal{E}} \; v_1$, then there exists $\mathcal{E}'$ such that $\mathcal{E} \subseteq \mathcal{E}'$ and $t_0\{v_0/x\} \; \mathcal{X}_{\mathcal{E}'} \; t_1\{v_1/x\}$.

Proposition 6.10. *If $\mathcal{X}$ is an environmental bisimulation up to environment, then $\mathcal{X} \subseteq \approx$.*

PROOF. As in [74].

In the following example we use the notion of a big-step environmental bisimulation up to environment that can be defined in an expected way by combining Definition 6.3 and 6.9.

Example 6.11 (double reset). We can simplify the proof of Example 6.7 by showing that for a given t the relation

$$\{(\emptyset, \langle\langle t\rangle\rangle, \langle t\rangle)\} \cup \{(\mathcal{I}, t', t') \mid t' \in \mathcal{T}_c\} \cup \{\mathcal{I}\}$$

is a big-step bisimulation up to environment.

Next, we define bisimulation up-to bisimilarity that allows one to simplify definitions of candidate relations by factoring out inessential bisimilar terms.

Definition 6.12. A relation $\mathcal{X}$ is an environmental bisimulation up to bisimilarity if

1. $t_0 \; \mathcal{X}_{\mathcal{E}} \; t_1$ implies:
   (a) if $t_0 \rightarrow_{\text{v}} t_0'$, then there exists t_1' such that $t_1 \rightarrow_{\text{v}}^* t_1'$ and $t_0' \; \mathcal{X}_{\mathcal{E}}\mathbb{E} \; t_1'$;
   (b) if t_0 is a stuck term $E_0[\mathcal{S}k.t_0']$, then there exists $E_1[\mathcal{S}k.t_1']$ such that $t_1 \Downarrow_{\text{v}} E_1[\mathcal{S}k.t_1]$, and for all $E_0' \; \ddot{\mathcal{E}} \; E_1'$, $\langle t_0'\{\lambda x.\langle E_0'[E_0[x]]\rangle/k\}\rangle \; \mathcal{X}_{\mathcal{E}}\mathbb{E} \; \langle t_1'\{\lambda x.\langle E_1'[E_1[x]]\rangle/k\}\rangle$ holds for a fresh x;
   (c) if t_0 is a value, then there exist v_1, v_1' such that $t_1 \rightarrow_{\text{v}}^* v_1$, $v_1' \; \mathbb{E} \; v_1$, and $\mathcal{E} \cup \{(t_0, v_1')\} \in \mathcal{X}$;
   (d) the converse of the above conditions on t_1;
2. for all $\mathcal{E} \in \mathcal{X}$, if $\lambda x.t_0 \; \mathcal{E} \; \lambda x.t_1$ and $v_0 \; \dot{\mathcal{E}} \; v_1$, then $t_0\{v_0/x\} \; \mathbb{E}\mathcal{X}_{\mathcal{E}}\mathbb{E} \; t_1\{v_1/x\}$.

Proposition 6.13. *If $\mathcal{X}$ is an environmental bisimulation up to bisimilarity, then $\mathcal{X} \subseteq \approx$.*

PROOF. As in [74].

As usual for up-to bisimilarity with small-step relations, we cannot compose on the left-hand side of $\mathcal{X}$ in clause (1) of Definition 6.12 [74].

6.2. Soundness and Completeness

We now prove soundness and completeness of $\mathbb{E}$ w.r.t. contextual equivalence. Because the proofs follow the same steps as for the λ-calculus [74], we only give here the main lemmas and sketch their proofs. The complete proofs can be found in Appendix C.1. For a relation $\mathcal{R}$ on terms, we write $\mathcal{R}^{\mathrm{v}}$ for its restriction to closed values. The first step consists in proving compatibility for values, and also for any terms but only w.r.t. evaluation contexts.

Lemma 6.14. *If $v_0 \approx_{\mathcal{E}} v_1$, then $C[v_0] \approx_{\mathcal{E}} C[v_1]$.*

Lemma 6.15. *If $t_0 \approx_{\mathcal{E}} t_1$, then $F[t_0] \approx_{\mathcal{E}} F[t_1]$.*

Lemmas 6.14 and 6.15 are proved simultaneously by showing that, for any environmental bisimulation $\mathcal{Y}$, the relation

$$\mathcal{X} \overset{\mathsf{def}}{=} \{(\dot{\mathcal{E}}^{\mathrm{v}}, F_0[t_0], F_1[t_1]) \mid t_0 \; \mathcal{Y}_{\mathcal{E}} \; t_1, F_0 \; \ddot{\mathcal{E}} \; F_1\} \\ \cup \{(\dot{\mathcal{E}}^{\mathrm{v}}, t_0, t_1) \mid \mathcal{E} \in \mathcal{Y}, t_0 \; \dot{\mathcal{E}} \; t_1\} \cup \{\dot{\mathcal{E}}^{\mathrm{v}} \mid \mathcal{E} \in \mathcal{Y}\}$$

is a bisimulation up-to environment. Informally, the elements of the first set of $\mathcal{X}$ reduce to elements of the second set of $\mathcal{X}$, and we then prove the bisimulation property for these elements by induction on $t_0 \; \dot{\mathcal{E}} \; t_1$. We can then prove the main compatibility lemma.

Theorem 6.16. *If $t_0 \; \mathbb{E} \; t_1$, then $C[t_0] \approx_{\dot{\mathbb{E}}^{\mathrm{v}}} C[t_1]$.*

We show that $\{(\dot{\mathbb{E}}^{\mathrm{v}}, t_0, t_1) \mid t_0 \; \dot{\mathbb{E}} \; t_1\} \cup \{\dot{\mathbb{E}}^{\mathrm{v}}\}$ is a bisimulation up-to bisimilarity by induction on $t_0 \; \dot{\mathbb{E}} \; t_1$. By weakening (Proposition 6.5), we can deduce from Theorem 6.16 that $\mathbb{E}$ is a compatible and, therefore, is sound w.r.t. $\mathbb{C}$.

Corollary 6.17 (Soundness). *We have $\mathbb{E} \subseteq \mathbb{C}$.*

The relation $\mathbb{E}$ is also complete w.r.t. contextual equivalence.

Theorem 6.18 (Completeness). *We have $\mathbb{C} \subseteq \mathbb{E}$.*

Proof. We prove that $\mathcal{X} \overset{\mathsf{def}}{=} \{(\mathbb{C}^{\mathrm{v}}, t_0, t_1) \mid t_0 \; \mathbb{C} \; t_1\} \cup \{\mathbb{C}^{\mathrm{v}}\}$ is a big-step environmental bisimulation.

Let $t_0 \; \mathcal{X}_{\mathbb{C}^{\mathrm{v}}} \; t_1$. If $t_0 \Downarrow_{\mathrm{v}} v_0$, then by definition of $\mathbb{C}$, there exists v_1 such that $t_1 \Downarrow_{\mathrm{v}} v_1$. By Proposition 6.6, we have $t_0 \; \mathbb{E} \; v_0$ and $t_1 \; \mathbb{E} \; v_1$, which implies $t_0 \; \mathbb{C} \; v_0$ and $t_1 \; \mathbb{C} \; v_1$ by Corollary 6.17. Transitivity of $\mathbb{C}$ gives $v_0 \; \mathbb{C} \; v_1$, hence we have $\mathbb{C}^{\mathrm{v}} \cup \{(v_0, v_1)\} = \mathbb{C}^{\mathrm{v}} \in \mathcal{X}$, as wished.

If $t_0 \Downarrow_{\mathrm{v}} E_0[\mathcal{S}k.t'_0]$, then by definition of $\mathbb{C}$, there exists $E_1[\mathcal{S}k.t'_1]$ such that $t_1 \Downarrow_{\mathrm{v}} E_1[\mathcal{S}k.t'_1]$. Let $E'_0 \; \ddot{\mathbb{C}}^{\mathrm{v}} \; E'_1$. By congruence of $\mathbb{C}$, we have $\langle E'_0[E_0[\mathcal{S}k.t'_0]]\rangle \; \mathbb{C} \; \langle E'_1[E_1[\mathcal{S}k.t'_1]]\rangle$. Because we have $\langle E'_0[E_0[\mathcal{S}k.t'_0]]\rangle \rightarrow_{\mathrm{v}} \langle t'_0\{\lambda x.\langle E'_0[E_0[x]]\rangle/k\}\rangle$, $\langle E'_1[E_1[\mathcal{S}k.t'_1]]\rangle \rightarrow_{\mathrm{v}} \langle t'_1\{\lambda x.\langle E'_1[E_1[x]]\rangle/k\}\rangle$, and also $\rightarrow_{\mathrm{v}} \subseteq \mathbb{E} \subseteq \mathbb{C}$, we obtain $\langle t'_0\{\lambda x.\langle E'_0[E_0[x]]\rangle/k\}\rangle \; \mathbb{C} \; \langle t'_1\{\lambda x.\langle E'_1[E_1[x]]\rangle/k\}\rangle$ (using transitivity of $\mathbb{C}$), which in turn implies $\langle t'_0\{\lambda x.\langle E'_0[E_0[x]]\rangle/k\}\rangle \; \mathcal{X}_{\mathbb{C}^{\mathrm{v}}} \; \langle t'_1\{\lambda x.\langle E'_1[E_1[x]]\rangle/k\}\rangle$, as required.

Let $\lambda x.t_0 \; \mathbb{C} \; \lambda x.t_1$ and $v_0 \; \dot{\mathbb{C}}^{\mathrm{v}} \; v_1$. By congruence of $\mathbb{C}$, we have $v_0 \; \mathbb{C} \; v_1$, and also $(\lambda x.t_0) \; v_0 \; \mathbb{C} \; (\lambda x.t_1) \; v_1$. Because $(\lambda x.t_0) \; v_0 \rightarrow_{\mathrm{v}} t_0\{v_0/x\}$, $(\lambda x.t_1) \; v_1 \rightarrow_{\mathrm{v}} t_1\{v_1/x\}$, and $\rightarrow_{\mathrm{v}} \subseteq \mathbb{E} \subseteq \mathbb{C}$, we have $t_0\{v_0/x\} \; \mathbb{C} \; t_1\{v_1/x\}$, i.e., $t_0\{v_0/x\} \; \mathcal{X}_{\mathbb{C}^{\mathrm{v}}} \; t_1\{v_1/x\}$, as wished.

6.3. Bisimulation up to context

Equivalence proofs based on environmental bisimilarity can be simplified by using up-to techniques, such as up to reduction, up to expansion, and up to context [74]. We only discuss the last, since the first two can be defined and proved sound in $\lambda_{\mathcal{S}}$ without issues. Bisimulations up to context may factor out a common context from the tested terms. Formally, two terms t_0, t_1 are related by the context closure of $\mathcal{X}$, written $t_0 \; \overline{\mathcal{X}_{\mathcal{E}}} \; t_1$, if

- either $t_0 = F_0[t_0']$, $t_1 = F_1[t_1']$, $t_0' \; \mathcal{X}_{\mathcal{E}} \; t_1'$, and $F_0 \; \ddot{\mathcal{E}} \; F_1$;
- or $t_0 \; \dot{\mathcal{E}} \; t_1$.

We then define environmental bisimulation up to context as follows.

Definition 6.19. A relation $\mathcal{X}$ is a bisimulation up to context if

1. $t_0 \; \mathcal{X}_{\mathcal{E}} \; t_1$ implies:
   (a) if $t_0 \to_{\mathrm{v}} t_0'$, then there exists t_1' such that $t_1 \to_{\mathrm{v}}^* t_1'$ and $t_0' \; \overline{\mathcal{X}_{\mathcal{E}}} \; t_1'$;
   (b) if t_0 is a stuck term $E_0[\mathcal{S}k.t_0']$, then there exists $E_1[\mathcal{S}k.t_1']$ such that $t_1 \Downarrow_{\mathrm{v}} E_1[\mathcal{S}k.t_1']$ and for all $E_0' \; \ddot{\mathcal{E}} \; E_1'$, $\langle t_0'\{\lambda x.\langle E_0'[E_0[x]]\rangle/k\}\rangle \; \overline{\mathcal{X}_{\mathcal{E}}} \; \langle t_1'\{\lambda x.\langle E_1'[E_1[x]]\rangle/k\}\rangle$ holds for a fresh x;
   (c) if t_0 is a value, then there exist v_1 and $\mathcal{E}' \in \mathcal{X}$ such that $t_1 \Downarrow_{\mathrm{v}} v_1$ and $\mathcal{E} \cup \{(t_0, v_1)\} \subseteq \dot{\mathcal{E}'}^{\mathrm{v}}$;
   (d) the converse of the above conditions on t_1;
2. for all $\mathcal{E} \in \mathcal{X}$, if $\lambda x.t_0 \; \mathcal{E} \; \lambda x.t_1$ and $v_0 \; \dot{\mathcal{E}} \; v_1$, then $t_0\{v_0/x\} \; \overline{\mathcal{X}_{\mathcal{E}}} \; t_1\{v_1/x\}$.

Proposition 6.20. *If $\mathcal{X}$ is a bisimulation up to context, then $\mathcal{X} \subseteq \approx$.*

Proof. As in [74].

Note that in the definition of context closure $\overline{\mathcal{X}_{\mathcal{E}}}$, terms related by $\mathcal{X}_{\mathcal{E}}$ can be put into evaluation contexts, while values (related by $\mathcal{E}$) can be put in any contexts. This restriction to evaluation contexts in the first case is usual in the definition of up-to context techniques for environmental relations [74, 83, 76, 66], and is in fact necessary for the technique to be sound, as pointed out by Madiot [58]. Indeed, assume terms can be put in any context, and let $\mathcal{X} \stackrel{\mathsf{def}}{=} \{(\mathcal{E}, \Omega, \Omega), (\mathcal{E}, (\lambda x.x) \; (\lambda y.\Omega), (\lambda x.x) \; (\lambda y.\Omega))\}$ where $\mathcal{E}$ is any environment. Then $\mathcal{X}$ is a bisimulation up to context, because $(\lambda x.x)\,(\lambda y.\Omega) \to_{\mathrm{v}} \lambda y.\Omega$, and $(\lambda y.\Omega) \; \dot{\mathcal{X}}_{\mathcal{E}} \; (\lambda y.\Omega)$ holds, since $\Omega \; \mathcal{X}_{\mathcal{E}} \; \Omega$. Then Proposition 6.20 implies $(\lambda x.x)\,(\lambda y.\Omega) \approx_{\mathcal{E}} (\lambda x.x)\,(\lambda y.\Omega)$, which in turn implies $\mathcal{E} \cup \{(\lambda y.\Omega, \lambda y.\Omega)\} \subseteq \approx$ for any $\mathcal{E}$, including any unsound environment such as $\{(\lambda x.\Omega, \lambda x.x)\}$, hence a contradiction.

The above reasoning does not depend on the calculus, as it holds for the plain λ-calculus; environments are what makes the restriction to evaluation contexts necessary. This is not problematic for the λ-calculus, because when a term t reduces within an evaluation context, the context is not affected. It is not the case in $\lambda_{\mathcal{S}}$, as (a part of) the evaluation context can be captured. As a result, bisimulation up to context is not as helpful in $\lambda_{\mathcal{S}}$ as in the λ-calculus. To illustrate this, suppose we want to construct a candidate relation $\mathcal{X}$ to prove the β_Ω axiom, starting from $(\emptyset, E[t], (\lambda x.E[x])\, t)$ (with $x \notin \mathsf{fv}(E)$). The problematic case is when t is a stuck term $E_0[\mathcal{S}k.t_0]$: for $\mathcal{X}$ to be a bisimulation, we would have to add $(\emptyset, \langle t_0\{\lambda y.\langle E'[(\lambda x.E[x])\, E_0[y]]\rangle/k\}\rangle, \langle t_0\{\lambda y.\langle E'[E[E_0[y]]]\rangle/k\}\rangle)$ to $\mathcal{X}$ for all E'. At this point, we would like to use the up-to context technique, because the subterms $(\lambda x.E[x])\, E_0[y]$ and $E[E_0[y]]$ are similar to the terms we want to relate (they can be written $(\lambda x.E[x])\, t''$ and $E[t'']$ with $t'' = E_0[y]$). However, we have at best $\langle t_0\{\lambda y.\langle E_1[(\lambda x.E[x])\, E_0[y]]\rangle/k\}\rangle \; \dot{\mathcal{X}}_{\mathcal{E}} \; \langle t_0\{\lambda y.\langle E_2[E[E_0[y]]]\rangle/k\}\rangle$ (and not $\overline{\mathcal{X}_{\mathcal{E}}}$), because t_0 can be any term, so $(\lambda x.E[x])\, E_0[y]$ and $E[E_0[y]]$ can be put in any context, not necessarily in an evaluation one. Therefore, Definition 6.19 cannot be used here, a new kind of up-to technique is required. During the review process of this paper, we managed to define in [3] an environmental bisimulation up to context able to prove the β_Ω axiom for a richer calculus, using a completely different framework. We believe these ideas can be applied to $\lambda_{\mathcal{S}}$, and we leave this as a future work.

Note that we do not have such issues with normal-form bisimilarity (Section 4), because normal-form bisimulation up to context is not restricted to evaluation contexts only, since we do not have environments to cause problems. But even if environmental bisimulation up to context is not as helpful as wished, it still simplifies equivalence proofs, as we can see with the next examples, where the notion of a big-step environmental bisimulation is equipped with the up to context technique (we omit a formal definition that is a straightforward combination of Definition 6.3 and 6.19).

Example 6.21 (double reset). We can simplify further the proof of equivalence between $\langle t\rangle$ and $\langle\langle t\rangle\rangle$ by showing that

$$\{(\emptyset, \langle\langle t\rangle\rangle, \langle t\rangle), \emptyset\}$$

is a big-step bisimulation up to context. Indeed, the couple (v, v) belongs to the closure of the empty environment, so clause (1c) of Definition 6.19 is verified.

Example 6.22 (Turing's combinator). We prove that Θ is bisimilar to $\Theta_{\mathcal{S}}$ using bisimulation up to context. We define $\theta' \overset{\mathsf{def}}{=} \lambda x.\langle \theta\, x\rangle$, $v_0 \overset{\mathsf{def}}{=} \lambda y.y\,(\lambda z.\theta\,\theta\, y\, z)$, and $v_1 \overset{\mathsf{def}}{=} \lambda y.y\,(\lambda z.\theta'\,\theta'\, y\, z)$. We define $\mathcal{E}$ inductively by the following rules:

$$v_0 \;\mathcal{E}\; v_1 \qquad\qquad \frac{v \;\dot{\mathcal{E}}\; v'}{\lambda z.\theta\,\theta\, v\, z \;\mathcal{E}\; \lambda z.\theta'\,\theta'\, v'\, z}$$

Then $\mathcal{X} \overset{\mathsf{def}}{=} \{(\mathcal{E}, \Theta, \Theta_{\mathcal{S}}), (\mathcal{E}, \Theta, \theta'\,\theta'), \mathcal{E}\}$ is a (big-step) bisimulation up to context. Indeed, we have $\Theta \Downarrow_{\mathrm{v}} v_0$, $\Theta_{\mathcal{S}} \Downarrow_{\mathrm{v}} v_1$, and $\theta'\,\theta' \Downarrow_{\mathrm{v}} v_1$; therefore, clause (1c) of Definition 6.19 is checked for both pairs.

We now verify clause (2) on the environment $\mathcal{E}$. Let $v_0' \;\dot{\mathcal{E}}\; v_1'$. In the case $v_0 \;\mathcal{E}\; v_1$, we have to check that $v_0'\,(\lambda z.\theta\,\theta\, v_0'\, z) \;\overline{\mathcal{X}_{\mathcal{E}}}\; v_1'\,(\lambda z.\theta'\,\theta'\, v_1'\, z)$ holds. From $\lambda z.\theta\,\theta\, v_0'\, z \;\mathcal{E}\; \lambda z.\theta'\,\theta'\, v_1'\, z$ and $v_0' \;\dot{\mathcal{E}}\; v_1'$, we deduce $v_0'\,(\lambda z.\theta\,\theta\, v_0'\, z) \;\dot{\mathcal{E}}\; v_1'\,(\lambda z.\theta'\,\theta'\, v_1'\, z)$, and $\dot{\mathcal{E}} \subseteq \overline{\mathcal{X}_{\mathcal{E}}}$, hence the result holds. In the other subcase, we have $\lambda z.\theta\,\theta\, v\, z \;\mathcal{E}$

$\lambda z.\theta'\,\theta'\,v'\,z$ with $v\;\dot{\mathcal{E}}\;v'$. We have to check that $\theta\,\theta\,v\,v_0'\;\overline{\mathcal{X}_{\mathcal{E}}}\;\theta'\,\theta'\,v'\,v_1'$ holds. From $v\;\dot{\mathcal{E}}\;v'$ and $v_0'\;\dot{\mathcal{E}}\;v_1'$, we deduce $\Box\;v\;v_0'\;\ddot{\mathcal{E}}\;\Box\;v'\;v_1'$, and we have $\theta\,\theta\;\mathcal{X}_{\mathcal{E}}\;\theta'\,\theta'$, hence the result holds.

Bisimulation up to context can be combined with the other up-to techniques, such as up to reduction [74] (which allows extra reduction steps before relating the terms), as we can see with the following examples.

6.4. Proving the axioms

We discuss Kameyama and Hasegawa's axioms (Section 2.4) in the context of environmental bisimulation. The $\beta_{\mathtt{v}}$, $\langle\cdot\rangle_{\mathcal{S}}$, and $\langle\cdot\rangle_{\mathtt{val}}$ axioms are direct consequences of Proposition 6.6. We now prove the remaining axioms (except for $\mathcal{S}_{\mathtt{elim}}$, because we use the relaxed semantics).

Proposition 6.23 ($\eta_{\mathtt{v}}$ axiom). *If $x \notin \mathsf{fv}(v)$, then $\lambda x.v\,x\;\mathbb{E}\;v$.*

Proof. Let $\mathcal{E} \stackrel{\mathsf{def}}{=} \{(\lambda x.v\,x, v) \mid v \in \mathcal{T}_c, x \notin \mathsf{fv}(v)\}$. Then $\{(\emptyset, \lambda x.v\,x, v) \mid v \in \mathcal{T}_c, x \notin \mathsf{fv}(v)\} \cup \{\mathcal{E}\}$ is a bisimulation up to context up to reduction. Suppose $v = \lambda y.t$, and let $v_0\;\dot{\mathcal{E}}\;v_1$. We have to relate $v\,v_0$ and $t\{v_1/y\}$, but $v\,v_0 \rightarrow_{\mathrm{v}} t\{v_0/y\}\;\dot{\mathcal{E}}\;t\{v_1/y\}$, hence the result holds.

Proposition 6.24 ($\mathcal{S}_{\langle\cdot\rangle}$ axiom). *We have $\mathcal{S}k.\langle t\rangle\;\mathbb{E}\;\mathcal{S}k.t$.*

Proof. The relation $\mathcal{X} \stackrel{\mathsf{def}}{=} \{(\emptyset, \mathcal{S}k.\langle t\rangle, \mathcal{S}k.t), (\emptyset, \langle\langle t\rangle\rangle, \langle t\rangle) \mid t \in \mathcal{T}_c\} \cup \{\emptyset\}$ is a bisimulation up to context. Let $\mathcal{S}k.\langle t\rangle\;\mathcal{X}_\emptyset\;\mathcal{S}k.t$. We have to check that for all E, $\langle\langle t\{\lambda x.\langle E[x]\rangle/k\}\rangle\rangle\;\mathcal{X}_\emptyset\;\langle t\{\lambda x.\langle E[x]\rangle/k\}\rangle$ holds (for a fresh x), but we obtain two terms of the form $\langle\langle t'\rangle\rangle$ and $\langle t'\rangle$, which are related by $\mathcal{X}_\emptyset$ by definition of $\mathcal{X}$. Bisimulation up to context holds on the latter terms as pointed out in Example 6.21.

Proposition 6.25 ($\langle\cdot\rangle_{\mathtt{lift}}$ axiom). *We have $\langle(\lambda x.t_0)\,\langle t_1\rangle\rangle\;\mathbb{E}\;(\lambda x.\langle t_0\rangle)\,\langle t_1\rangle$.*

Proof. The relation $\{(\emptyset, \langle(\lambda x.t)\,\langle t'\rangle\rangle, (\lambda x.\langle t\rangle)\langle t'\rangle), \emptyset\}$ is a big-step bisimulation up to context (the argument is the same as for Example 6.21).

Proposition 6.26 (β_Ω axiom). *If $x \notin \mathsf{fv}(E)$, then $(\lambda x.E[x])\,t\;\mathbb{E}\;E[t]$.*

Proof. The bisimulation $\mathcal{R}$ defined in the proof of Proposition 5.23 can be changed into an environmental relation $\mathcal{X}$ by considering that the terms in $\mathcal{R}$ are related with the empty environment $\emptyset$. We can then prove that $\mathcal{X}$ is an environmental bisimulation up to context, following the same scheme as in the proof of Proposition 5.23.

6.5. Environmental Relations for the Original Semantics

The bisimilarities introduced so far are sound and complete w.r.t. the contextual equivalence $\mathbb{C}$ of the relaxed semantics, but only sound w.r.t. the contextual equivalence $\mathbb{P}$ of the original semantics (cf. Proposition 3.9). We now propose a definition of environmental bisimulation adapted to delimited terms (but defined on all terms, like $\mathbb{P}$).

Definition 6.27. A relation $\mathcal{X}$ is a delimited environmental bisimulation if

1. if $t_0 \; \mathcal{X}_\mathcal{E} \; t_1$ and t_0 and t_1 are not both delimited terms, then for all closed E_0, E_1 such that $E_0 \; \ddot{\mathcal{E}} \; E_1$, we have $\langle E_0[t_0]\rangle \; \mathcal{X}_\mathcal{E} \; \langle E_1[t_1]\rangle$;
2. $p_0 \; \mathcal{X}_\mathcal{E} \; p_1$ implies
   - (a) if $p_0 \rightarrow_\mathrm{v} p_0'$, then there exists p_1' such that $p_1 \rightarrow_\mathrm{v}^* p_1'$ and $p_0' \; \mathcal{X}_\mathcal{E} \; p_1'$;
   - (b) if $p_0 \rightarrow_\mathrm{v} v_0$, then there exists v_1 such that $p_1 \rightarrow_\mathrm{v}^* v_1$, and $\{(v_0, v_1)\} \cup \mathcal{E} \in \mathcal{X}$;
   - (c) the converse of the above conditions on p_1;
3. for all $\mathcal{E} \in \mathcal{X}$, if $\lambda x.t_0 \; \mathcal{E} \; \lambda x.t_1$ and $v_0 \; \dot{\mathcal{E}} \; v_1$, then $t_0\{v_0/x\} \; \mathcal{X}_\mathcal{E} \; t_1\{v_1/x\}$.

Delimited environmental bisimilarity, written $\simeq$, is the largest delimited environmental bisimulation. As before, the relation $\simeq_\emptyset$, also written $\mathbb{F}$, is candidate to characterize $\mathbb{P}$.

Clauses (2) and (3) of Definition 6.27 deal with delimited terms and environments in a classical way (as in plain λ-calculus). The problematic case is when relating terms t_0 and t_1 that are not both delimited terms (clause (1)). Indeed, one of them may be control stuck and, therefore, we have to test them within some contexts $\langle E_0\rangle$, $\langle E_1\rangle$ (built from $\mathcal{E}$) to potentially trigger a capture that otherwise would not happen. We cannot require both terms to be control stuck, as in clause (1b) of Definition 6.1, because a control stuck term can be equivalent to a term free from control effect. E.g., we will see that $v \; \mathbb{F} \; \mathcal{S}k.k \, v$, provided that $k \notin \mathsf{fv}(v)$.

Example 6.28. Suppose we want to prove $\langle(\lambda x.t_0) \, \langle t_1\rangle\rangle \; \mathbb{F} \; (\lambda x.\langle t_0\rangle) \, \langle t_1\rangle$ (as in Proposition 6.25). Because $(\lambda x.\langle t_0\rangle) \, \langle t_1\rangle$ is not delimited, we have to put both terms into a context first: we have to change the candidate relation of Proposition 6.25 into

$$\begin{array}{l}\{(\emptyset, \langle(\lambda x.t_0) \, \langle t_1\rangle\rangle, (\lambda x.\langle t_0\rangle) \, \langle t_1\rangle), \emptyset\} \\ \cup \\ \{(\emptyset, \langle E[\langle(\lambda x.t_0) \, \langle t_1\rangle\rangle]\rangle, \langle E[(\lambda x.\langle t_0\rangle) \, \langle t_1\rangle]\rangle) \mid E \in \mathcal{PC}_c\}.\end{array}$$

In contrast, to prove $\langle\langle t\rangle\rangle \; \mathbb{F} \; \langle t\rangle$, we do not have to change the candidate relation of Example 6.7, since both terms are delimited.

We can give a definition of big-step bisimulation by removing clause (2a) and changing $\rightarrow_\mathrm{v}$ into $\Downarrow_\mathrm{v}$ in clause (2b). Propositions 6.5 and 6.6 can also be extended to $\simeq$ and $\mathbb{F}$.

Proposition 6.29. *If $t_0 \simeq_\mathcal{E} t_1$ and $\mathcal{E}' \subseteq \mathcal{E}$ then $t_0 \simeq_{\mathcal{E}'} t_1$.*

Proposition 6.30. *If $t_0 \rightarrow_{\mathrm{v}} t'_0$, then $t_0 \ \mathbb{F} \ t'_0$.*

The next proposition states that $\mathbb{E}$ is more discriminating than $\mathbb{F}$.

Proposition 6.31. *We have $\mathbb{E} \subseteq \mathbb{F}$.*

A consequence of Proposition 6.31 is that we can use Definition 6.1 as a proof technique for $\mathbb{F}$. E.g., we have directly $\langle(\lambda x.t_0)\ \langle t_1\rangle\rangle \ \mathbb{F} \ (\lambda x.\langle t_0\rangle)\ \langle t_1\rangle$, because $\langle(\lambda x.t_0)\ \langle t_1\rangle\rangle \ \mathbb{E} \ (\lambda x.\langle t_0\rangle)\ \langle t_1\rangle$.

6.6. Soundness and Completeness

We sketch the proofs of soundness and completeness of $\mathbb{F}$ w.r.t. $\mathbb{P}$; see Appendix C.2 for the complete proofs. The soundness proof follows the same scheme as in Section 6.2, with some necessary adjustments. As before, we need up-to environment and up-to bisimilarity techniques to prove the following lemmas.

Lemma 6.32. *If $v_0 \simeq_{\mathcal{E}} v_1$, then $C[v_0] \simeq_{\mathcal{E}} C[v_1]$.*

Lemma 6.33. *If $t_0 \simeq_{\mathcal{E}} t_1$, then $F[t_0] \simeq_{\mathcal{E}} F[t_1]$.*

We prove Lemmas 6.32 and 6.33 by showing that a relation similar to the relation $\mathcal{X}$ defined in Section 6.2 is a bisimulation up to environment. We then want to prove the main compatibility lemma, akin to Theorem 6.16, by showing that $\mathcal{Y} \stackrel{\mathsf{def}}{=} \{(\dot{\mathbb{F}}^{\mathrm{v}}, t_0, t_1) \mid t_0 \ \dot{\mathbb{F}} \ t_1\} \cup \{\dot{\mathbb{F}}^{\mathrm{v}}\}$ is a bisimulation up to bisimilarity. However, we can no longer proceed by induction on $t_0 \ \dot{\mathbb{F}} \ t_1$, as for Theorem 6.16. Indeed, if $p_0 = \langle t_0\rangle$, $p_1 = \langle t_1\rangle$ with $t_0 \ \dot{\mathbb{F}} \ t_1$, and if t_0 is a control stuck term, then p_0 reduces to some term, but the induction hypothesis does not tell us anything about t_1. To circumvent this, we decompose related delimited terms into related subcomponents.

Lemma 6.34. *If $p_0 \ \dot{\mathbb{F}} \ p_1$, then either $p_0 \ \mathbb{F} \ p_1$, or one of the following holds:*

- *$p_0 = \langle v_0\rangle$;*
- *$p_0 = F_0[\langle E_0[t_0]\rangle]$, $p_1 = F_1[\langle E_1[t_1]\rangle]$, $F_0 \ \ddot{\mathbb{F}} \ F_1$, $E_0 \ \ddot{\mathbb{F}} \ E_1$, $t_0 \ \mathbb{F} \ t_1$ and $t_0 \rightarrow_{\mathrm{v}} t'_0$ or t_0 is stuck;*
- *$p_0 = F_0[\langle E_0[r_0]\rangle]$, $p_1 = F_1[\langle E_1[t_1]\rangle]$, $F_0 \ \ddot{\mathbb{F}} \ F_1$, $E_0 \ \ddot{\mathbb{F}} \ E_1$, $r_0 \ \dot{\mathbb{F}} \ t_1$ but $r_0 \ \not{\mathbb{F}} \ t_1$.*

Lemma 6.34 generalizes Proposition 2.7 to related delimited terms: we know p_0 can be decomposed into contexts F, $\langle E\rangle$, and a redex r, and we relate these subterms to p_1. We can then prove that $\mathcal{Y}$ (defined above) is a bisimulation up to bisimilarity, by showing that, in each case described by Lemma 6.34, p_0 and p_1 reduce to terms related by $\mathcal{Y}$. From this, we deduce $\mathbb{F}$ is a congruence, and is sound w.r.t. $\mathbb{P}$.

Theorem 6.35. *If $t_0 \ \mathbb{F} \ t_1$, then $C[t_0] \simeq_{\dot{\mathbb{F}}^{\mathrm{v}}} C[t_1]$.*

Corollary 6.36 (Soundness). *We have* $\mathbb{F} \subseteq \mathbb{P}$.

Remark 6.37. Following the ideas behind Definition 6.27, one can define an applicative bisimilarity $\mathbb{B}$ for delimited terms. A relation $\mathcal{R}$ on closed terms is a delimited applicative simulation if the following conditions are met:

- if $t_0 \mathrel{\mathcal{R}} t_1$ and t_0 and t_1 are not both delimited terms, then for all closed E, we have $\langle E[t_0]\rangle \mathrel{\mathcal{R}} \langle E[t_1]\rangle$;
- $p_0 \mathrel{\mathcal{R}} p_1$ implies
  - if $p_0 \rightarrow_{\mathrm{v}} p_0'$, then there exists p_1' such that $p_1 \rightarrow_{\mathrm{v}}^* p_1'$ and $p_0' \mathrel{\mathcal{R}} p_1'$;
  - if $p_0 \rightarrow_{\mathrm{v}} \lambda x.t_0$, then there exists t_1 such that $p_1 \rightarrow_{\mathrm{v}}^* \lambda x.t_1$, and for all v, we have $t_0\{v/x\} \mathrel{\mathcal{R}} t_1\{v/x\}$.

A relation $\mathcal{R}$ is a delimited applicative bisimulation if $\mathcal{R}$ and $\mathcal{R}^{-1}$ are delimited applicative simulation. Delimited applicative bisimilarity $\mathbb{B}$ is the largest delimited applicative bisimulation.

We conjecture $\mathbb{B}$ to be sound w.r.t. $\mathbb{P}$, but we do not know how to apply Howe's method to $\mathbb{B}$. To prove that the Howe's closure $\mathbb{B}^\bullet$ is a delimited applicative bisimulation, we need a version of Lemma 6.34 which decomposes terms related by $\mathbb{B}^\bullet$. The clause $\mathbb{B}^\bullet\mathbb{B} \subseteq \mathbb{B}^\bullet$ in the definition of the Howe's closure makes writing such a decomposition lemma difficult. We believe a new proof technique is necessary to show the soundness of $\mathbb{B}$.

We prove that $\mathbb{F}$ is complete by showing that $\mathbb{P}$ is a big-step bisimulation.

Theorem 6.38 (Completeness). *We have* $\mathbb{P} \subseteq \mathbb{F}$.

Proof. We prove that $\mathcal{X} = \{(\mathbb{P}^{\mathrm{v}}, t_0, t_1) \mid t_0 \mathrel{\mathbb{P}} t_1\} \cup \{\mathbb{P}^{\mathrm{v}}\}$ is a delimited big-step environmental bisimulation.

Let $t_0 \mathrel{\mathcal{X}_{\mathbb{P}^{\mathrm{v}}}} t_1$ such that t_0 and t_1 are not both delimited terms. Because $\mathbb{P}$ is a congruence, we have $\langle E_0[t_0]\rangle \mathrel{\mathbb{P}} \langle E_1[t_1]\rangle$ for all $E_0 \mathrel{\ddot{\mathbb{P}}^{\mathrm{v}}} E_1$, i.e., $\langle E_0[t_0]\rangle \mathrel{\mathcal{X}_{\mathbb{P}^{\mathrm{v}}}} \langle E_1[t_1]\rangle$ as wished.

Let $p_0 \mathrel{\mathcal{X}_{\mathbb{P}^{\mathrm{v}}}} p_1$ such that $p_0 \rightarrow_{\mathrm{v}}^* v_0$. We have $\langle p_0\rangle \rightarrow_{\mathrm{v}}^* v_0$, so by definition of $\mathbb{P}$, there exists v_1 such that $\langle p_1\rangle \rightarrow_{\mathrm{v}}^* v_1$, which implies $p_1 \rightarrow_{\mathrm{v}}^* v_1$ (because p_1 is delimited). By Proposition 6.30, we have $p_0 \mathrel{\mathbb{F}} v_0$ and $p_1 \mathrel{\mathbb{F}} v_1$, which implies $p_0 \mathrel{\mathbb{P}} v_0$ and $p_1 \mathrel{\mathbb{P}} v_1$ by soundness of $\mathbb{F}$. Transitivity of $\mathbb{P}$ gives $v_0 \mathrel{\mathbb{P}} v_1$, hence we have $\mathbb{P}^{\mathrm{v}} \cup \{(v_0, v_1)\} = \mathbb{P}^{\mathrm{v}} \in \mathcal{X}$, as wished.

Let $\lambda x.t_0 \mathrel{\mathbb{P}} \lambda x.t_1$ and $v_0 \mathrel{\dot{\mathbb{P}}^{\mathrm{v}}} v_1$. By congruence of $\mathbb{P}$, we have $v_0 \mathrel{\mathbb{P}} v_1$, and also $(\lambda x.t_0)\, v_0 \mathrel{\mathbb{P}} (\lambda x.t_1)\, v_1$. Because $(\lambda x.t_0)\, v_0 \rightarrow_{\mathrm{v}} t_0\{v_0/x\}$, $(\lambda x.t_1)\, v_1 \rightarrow_{\mathrm{v}} t_1\{v_1/x\}$ and $\rightarrow_{\mathrm{v}} \subseteq \mathbb{F} \subseteq \mathbb{P}$, we have $t_0\{v_0/x\} \mathrel{\mathbb{P}} t_1\{v_1/x\}$, i.e., $t_0\{v_0/x\} \mathrel{\mathcal{X}_{\mathbb{P}^{\mathrm{v}}}} t_1\{v_1/x\}$, as wished.

We can also define up-to techniques adapted to delimited terms, but we have the same limitations as in Section 6.3 for bisimulation up to context.

6.7. Examples

Because delimited environmental bisimilarity is sound and complete, and delimited normal-form bisimulation is sound, any equivalence proved in Section 4.7 also holds with delimited environmental bisimilarity; in particular, we also have $\equiv \subseteq \mathbb{F}^\circ$. However, as in relaxed semantics, the lack of tractable enough environmental bisimulation up to context makes the equivalence proofs more difficult with environmental than with normal-form bisimulation. Combining the proof techniques can help, e.g., for the $\mathcal{S}_{\mathtt{elim}}$ axiom.

Proposition 6.39 ($\mathcal{S}_{\mathtt{elim}}$ axiom). *If $k \notin \mathsf{fv}(t)$, then $t \; \mathbb{F} \; \mathcal{S}k.k\, t$.*

The relation $\{(\emptyset, t, \mathcal{S}k.k\,t), (\emptyset, \langle E[t]\rangle, \langle E[\mathcal{S}k.k\,t]\rangle) \mid t \in \mathcal{T}, E \in \mathcal{PC}, k \notin \mathsf{fv}(t)\} \cup \mathbb{F}$ is a delimited bisimulation. We have $\langle E[\mathcal{S}k.k\,t]\rangle \rightarrow_{\mathrm{v}} \langle(\lambda x.\langle E[x]\rangle)\,t\rangle$, and from Example 4.17, we know that $\langle(\lambda x.\langle E[x]\rangle)\,t\rangle \; \mathbb{N} \; \langle E[t]\rangle$ holds. Because we have $\mathbb{N} \subseteq \mathbb{C} = \mathbb{E} \subseteq \mathbb{F}$, we can deduce $\langle(\lambda x.\langle E[x]\rangle)\,t\rangle \; \mathbb{F} \; \langle E[t]\rangle$. Without Example 4.17, the proof would be as difficult as the proof of the next result, which is also taken from Section 4.7.

Proposition 6.40. *If $k \notin \mathsf{fv}(t_1)$, then $(\lambda x.\mathcal{S}k.t_0)\,t_1 \; \mathbb{F} \; \mathcal{S}k.((\lambda x.t_0)\,t_1)$.*

Putting the terms above in a context $\langle E\rangle$, we want to relate $\langle E[(\lambda x.\mathcal{S}k.t_0)\,t_1]\rangle$ with $\langle E[\mathcal{S}k.((\lambda x.t_0)\,t_1)]\rangle$. The second term reduces to $\langle(\lambda x.t_0\{\lambda y.\langle E[y]\rangle/k\})\,t_1\rangle$, so we can conclude by showing $\langle E[(\lambda x.\mathcal{S}k.t_0)\,t_1]\rangle \; \mathbb{F} \; \langle(\lambda x.t_0\{\lambda y.\langle E[y]\rangle/k\})\,t_1\rangle$. If t_1 performs successive captures, we can obtain terms like

$$\begin{array}{l}\langle E[(\lambda x.\mathcal{S}k.t_0)\,F[\langle E[(\lambda x.\mathcal{S}k.t_0)\,t_1']\rangle]]\rangle \text{ and}\\ \langle(\lambda x.t_0\{\lambda y.\langle E[y]\rangle/k\})\,F[\langle(\lambda x.t_0\{\lambda y.\langle E[y]\rangle/k\})\,t_1'\rangle]\rangle\end{array}$$

with nested contexts. We define inductively two sequences of terms which represent all these possible nestings, and we prove they are delimited bisimilar. The complete proof can be found in Appendix C.3.

6.8. Conclusion

In the relaxed semantics, the environmental bisimilarity $\mathbb{E}$ tests λ-abstractions and control stuck terms by, respectively, applying them to values, and putting them into a pure evaluation context (surrounded by a reset). The testing arguments and contexts are generated from an environment, which represents the current knowledge of an outside observer about the tested terms. The resulting relation is sound and complete w.r.t. $\mathbb{C}$. Plain environmental bisimulation is harder to use than applicative bisimulation, but it is supposed to be used in conjunction with up-to techniques. With up-to techniques (and their combination, as in Section 6.4), the proofs can be simpler than with applicative bisimilarity (compare, e.g., Example 6.22 with Example 5.9), and the candidate relations can be very small (see, e.g., Example 6.21). However, the bisimulation up to context that we define here is not as helpful as we could hope (see Section 6.3), and cannot be used when proving, e.g., the β_Ω axiom. A different framework seems to be required, like the one we use in [3].

We also define an environmental bisimilarity $\mathbb{F}$ for the original semantics, which is complete w.r.t. the contextual equivalence $\mathbb{P}$. The difference with $\mathbb{E}$ is that we put terms that are not delimited into a context $\langle E\rangle$ before comparing them. Unlike applicative bisimilarity (see Remark 6.37), the soundness proof technique for the environmental bisimilarity is flexible enough to be adapted to this setting. Note that the definition of $\mathbb{F}$ (Definition 6.27) is only a small improvement over the definition of $\mathbb{P}$, as it contains a quantification over (pure) contexts for a large class of terms (the non-delimited terms), and the resulting equivalence proof technique is much more difficult to use than delimited normal-form bisimilarity (compare the proof of Proposition 4.41 to Appendix C.3). A future work would be to restrict the class of terms on which such a quantification over contexts is necessary.

7. Extensions

In this section, we discuss how our results are affected if we consider other semantics for $\lambda_{\mathcal{S}}$, or if we study other delimited-control operators.

7.1. Local Reduction Rules

In the semantics of Section 2, contexts are captured in one reduction step. Another usual way of computing capture is to use local reduction rules, where the context is consumed piece by piece [24]. Formally, we introduce *elementary contexts*, defined as follows:

$$\text{Elementary contexts:} \quad D \quad ::= \quad v\,\Box \ \mid\ \Box\, t$$

The reduction rule (*shift*) is then replaced with the next two rules.

$$\begin{array}{rll} F[D[\mathcal{S}k.t]] & \rightarrow_{\mathrm{v}} F[\mathcal{S}k'.t\{\lambda x.\langle k'\ D[x]\rangle/k\}] \text{ with } x, k' \notin \mathsf{fv}(D)\cup\mathsf{fv}(t) & (\mathit{shift}_D) \\ F[\langle \mathcal{S}k.t\rangle] & \rightarrow_{\mathrm{v}} F[\langle t\{\lambda x.x/k\}\rangle] & (\mathit{shift}_I) \end{array}$$

As we can see in rule (shift_D), the capture of an elementary context does not require a reset, and it leaves the operator shift in place to continue the capture process. The process stops when a reset is encountered, in which case the rule (shift_I) applies: the shift operator is removed, and its variable k is replaced with the function representing the delimited empty context.

Note that with local reduction rules, control stuck terms are of the form $\mathcal{S}k.t$ (without any surrounding context). This has major consequences on the definition of normal-form bisimulations. Indeed, Definition 4.1 extracts from control stuck terms their surrounding pure contexts and the bodies of their shift operator, and then relates these sub-components. Without surrounding contexts, normal-form bisimilarity can only compare the shift bodies. As a result, there is no point in distinguishing Definition 4.1 from refined bisimilarity (Definition 4.19): with local rules, two stuck terms $\mathcal{S}k.t_0$ and $\mathcal{S}k.t_1$ are normal-form bisimilar if $\langle t_0\rangle$ and $\langle t_1\rangle$ are normal-form bisimilar. The resulting bisimulation proofs are arguably more difficult than with the semantics of Section 2, as we can see with the next example.

Example 7.1 (β_Ω axiom). Assume we want to prove that $(\lambda x.E[x])\, t \;\mathbb{N}\; E[t]$ ($x \notin \mathsf{fv}(E)$) with local rules. If t is a control stuck term $\mathcal{S}k.t'$, we have to relate $\langle t'\{\lambda y.\langle k'\, (\lambda x.E[x])\, y\rangle/k\}\rangle$ (y, k' fresh) with $\langle t'\overrightarrow{\sigma}\rangle$, where $\overrightarrow{\sigma}$ are the substitutions we obtain as a result of the progressive capture of E by $\mathcal{S}k.t'$. To do so, one has to see what happens when k is applied to a value v (i.e., if $t' = F[k\, v]$). The proof is doable using bisimulation up to context, while the proof with the semantics of Section 2 uses only plain normal-form bisimulation (Proposition 4.31).

The theory for applicative and environmental bisimulations is not affected by using local rules; in particular, we still have to compare control stuck terms with a pure context E for these two relations. However, a proof using a small-step bisimulation of any kind becomes tedious with local rules. Indeed, these rules introduce a lot of redexes (first to capture a whole pure context, and then to reduce all the produced β-redexes), and a reduction of each redex has to be matched in a small-step relation. We, therefore, believe that the reduction rules of Section 2 are better suited to proving the equivalence of two $\lambda_{\mathcal{S}}$ terms.

7.2. Call-by-Name Reduction Semantics

In call-by-name, arguments are not reduced to values before β-reduction takes place. Such a semantics can be achieved by changing the syntax of (pure) evaluation contexts as follows:

$$\begin{array}{lrcl} \text{CBN pure contexts:} & E & ::= & \square \mid E\, t \\ \text{CBN evaluation contexts:} & F & ::= & \square \mid F\, t \mid \langle F\rangle \end{array}$$

and by turning the β-reduction rule into

$$F[(\lambda x.t_0)\, t_1] \quad \rightarrow_{\mathrm{n}} \quad F[t_0\{t_1/x\}] \qquad (\beta_n)$$

The rules (*shift*) and (*reset*) are the same as in call-by-value, but their meanings change because of the new syntax for call-by-name contexts. We still distinguish the relaxed semantics (without outermost enclosing reset) from the original semantics.

The results of this paper can be adapted to call-by-name by transforming values used as arguments into arbitrary terms, for example when comparing λ-abstractions with applicative bisimilarity, or when building testing terms from the environment in environmental bisimilarity. We can also relate the bisimilarities to the call-by-name CPS equivalence, which has been axiomatized by Kameyama and Tanaka [39]. The axioms for call-by-name are the same or simpler than in call-by-value: the axioms $\langle\cdot\rangle_{\mathcal{S}}$, $\langle\cdot\rangle_{\mathtt{val}}$, and $\mathcal{S}_{\langle\cdot\rangle}$ can be proved in call-by-name using bisimulations with the same proofs as in call-by-value. The call-by-value axioms β_{v}, β_Ω, and $\langle\cdot\rangle_{\mathtt{lift}}$ are replaced by a single axiom for call-by-name β-reduction

$$(\lambda x.t_0)\, t_1 =_{\mathrm{KT}} t_0\{t_1/x\},$$

which is straightforward to prove since the three bisimilarities contain reduction. Finally, the axiom $\mathcal{S}_{\mathtt{elim}}$ still holds only for the original semantics.

7.3. CPS-based Equivalences

It is possible to go beyond CPS equivalence and use the CPS definition of shift and reset to define behavioral equivalences in terms of it: t_0 and t_1 are bisimilar in $\lambda_{\mathcal{S}}$ if their translations $\overline{t_0}$ and $\overline{t_1}$ are bisimilar in the plain λ-calculus. As an example, we can define CPS applicative bisimilarity $\mathbb{A}_{\mathsf{CPS}}$ as follows: given two closed terms t_0 and t_1 of $\lambda_{\mathcal{S}}$, we have $t_0 \mathbb{A}_{\mathsf{CPS}} t_1$ if $\overline{t_0}$ and $\overline{t_1}$ are applicative bisimilar in the call-by-value λ-calculus [1]. We compare here this equivalence to the contextual equivalence $\mathbb{P}$ for the original semantics, since the CPS of Figure 1 is valid for that semantics only.

We conjecture that $\mathbb{A}_{\mathsf{CPS}}$ is sound w.r.t. $\mathbb{P}$, but we show it is not complete. A CPS translated term is of the form $\lambda k_1 k_2.t$, where k_1 and k_2 stand for, respectively, the continuation and the metacontinuation of the term, which are λ-abstractions of a special shape. But applicative bisimilarity in λ-calculus compares terms with any λ-abstraction, not just a continuation or metacontinuation, making $\mathbb{A}_{\mathsf{CPS}}$ over-discriminating compared to $\mathbb{P}$. Indeed, let $i \overset{\mathsf{def}}{=} \lambda y.y$, $v_0 \overset{\mathsf{def}}{=} \lambda x.\Omega$ and $v_1 \overset{\mathsf{def}}{=} \lambda x.\langle x\, i\rangle\, \Omega$. We have $v_0 \,\mathbb{P}\, v_1$, roughly because v_0 diverges as soon as it is applied to a value v, and so does v_1, either because $\langle v\, i\rangle$ diverges, or because $\langle v\, i\rangle$ converges and Ω then diverges (more formally, the relation $\{(\lambda x.\Omega, \lambda x.\langle x\, i\rangle\, \Omega)\} \cup \{(\Omega, \langle t\rangle\, \Omega) \mid t \in \mathcal{T}_c\} \cup \{(\Omega, v\, \Omega) \mid v \in \mathcal{V}_c\}$ is an applicative bisimulation, included in $\mathbb{C}$ and, therefore, in $\mathbb{P}$). The CPS translation of these terms, after some administrative reductions, yields

$$\overline{v_0} = \lambda k_1 k_2.k_1\, (\lambda x.\overline{\Omega})\, k_2, \text{ and}$$
$$\overline{v_1} = \lambda k_1 k_2.k_1\, (\lambda x k_1' k_2'.x\, (\lambda y.\overline{y})\, \gamma\, (\lambda z.\overline{\Omega}\, v'\, k_2))\, k_2$$

where γ is defined in Figure 1 and v' is some value, the precise definition of which is not important. If $v \overset{\mathsf{def}}{=} \lambda z k_2.z\, (\lambda x' k_1'' k_2''.i)\, i\, i$, then $\overline{v_0}\, v\, i \rightarrow_{\mathrm{v}}^{*} \overline{\Omega}$ and $\overline{v_1}\, v\, i \rightarrow_{\mathrm{v}}^{*} i$; the diverging part in $\overline{v_1}$, namely $\lambda z.\overline{\Omega}\, v'\, k_2$, is thrown away by v instead of being eventually applied, as it should be if v was a continuation (the term $\lambda x' k_1'' k_2''.i$ is not in the 2-layer CPS).

A possible way to get completeness for $\mathbb{A}_{\mathsf{CPS}}$ could be to restrict the target language of the CPS translation to a CPS calculus, i.e., a subcalculus where the grammar of terms enforces the correct shape of arguments passed as values, continuations, or metacontinuations (as in, e.g., [38]). However, even with completeness, we believe it is more tractable to work in direct style with the relations we define in this paper, than on CPS translations of terms: as we can see with v_0 and v_1 above, translating even relatively simple source terms leads to voluminous terms in CPS. Besides, $\mathbb{A}_{\mathsf{CPS}}$ compares all translated terms with a continuation (which corresponds to a context E) and a metacontinuation (which corresponds to a metacontext F), while bisimilarities in direct style need at most a context E to compare stuck terms.

Nonetheless, we believe that studying fully the relationship between CPS-based behavioral equivalences and direct-style equivalences is an interesting future work. We would like to consider other CPS translations, including a CPS translation for the relaxed semantics [60], or the 1-layer CPS translation for the original semantics [18]. We would also like to know if it is possible to obtain a CPS-based soundness proof for normal-form bisimilarity, as described at the beginning of Section 4.2, to have a complete picture of the interactions between CPS and behavioral equivalences.

7.4. The $\lambda\mu\widehat{\mathsf{tp}}$-Calculus

The $\lambda\mu$-calculus [65] contains a μ-construct that can be seen as an abortive control operator. In this calculus, we evaluate named terms of the form $[\alpha]t$, and the names α are used as placeholders for evaluation contexts. Roughly, a μ term $\mu\alpha.[\beta]t$ is able to capture its whole (named) evaluation context $[\gamma]E$, and substitutes α with $[\gamma]E$ in $[\beta]t$. Context substitution is the same as the one presented in Section 4.4. In particular, it is capture-free, e.g., in $[\beta]t\{[\gamma]E/\alpha\}$, the free names of $[\gamma]E$ (such as γ) cannot be bound by the μ constructs in t.

The $\lambda\mu\widehat{\mathsf{tp}}$-calculus [32] extends the $\lambda\mu$-calculus by adding a special name $\widehat{\mathsf{tp}}$ which can be dynamically bound during a context substitution. Besides, the μ-operator no longer captures the whole context, but only up to the nearest enclosing μ-binding of $\widehat{\mathsf{tp}}$. As a result, a μ-binding of $\widehat{\mathsf{tp}}$ can be seen as a delimiter, and in fact, the $\lambda\mu\widehat{\mathsf{tp}}$-calculus simulates $\lambda_{\mathcal{S}}$ [32]. In particular, their CPS equivalences coincide. However, defining bisimilarities in $\lambda\mu\widehat{\mathsf{tp}}$ may lead to relations similar to the $\lambda\mu$-calculus ones [14] because of names. Indeed, we have to compare named values $[\alpha]v$ in $\lambda\mu\widehat{\mathsf{tp}}$, which requires substituting α with some named context, as in the $\lambda\mu$-calculus [14]. Similarly, control stuck terms are of the form $[\alpha]F[\mu\widehat{\mathsf{tp}}.[\beta]v]$, and a way to relate them would be by replacing β with a context $[\gamma]E$ or $[\widehat{\mathsf{tp}}]E$. It would be interesting to compare the behavioral theories of $\lambda_{\mathcal{S}}$ and $\lambda\mu\widehat{\mathsf{tp}}$ to see if the encoding of the former into the latter is fully abstract (i.e., preserves contextual equivalence).

7.5. Typed Setting

Applicative, normal-form, and environmental bisimilarities have been studied in various typed settings, with different issues arising for each style of bisimilarity. In [29], Gordon points out that in a typed call-by-name λ-calculus, a context cannot distinguish a non-terminating term t from $\lambda x.t$ when they are both given a type of the form $A \to B$. As a result, the untyped definition of applicative bisimilarity is no longer suited in this case. Gordon fixed this issue in [29] by distinguishing passive from active types in its labeled transition system: a term with a passive type (such as a function type $A \to B$) can make a transition whether it converges or not, while terms of active type (such as int, $\ldots$) can make a transition only if it converges. Such a distinction is not necessary in call-by-value (all types are active). Applicative bisimilarity has also been defined for a typed object calculus in [30].

Normal-form bisimulation has been defined for a calculus with recursive types [55], a result then extended to existential types [56]. The difficulty here is to compare functions of type $A \to B$, when A is an empty type, such as $\mu X.\mathsf{bool} \times X$. These functions should be equivalent in call-by-value, and with applicative bisimilarity, we can relate them by simply arguing that there are no argument values of type A. But normal-form bisimilarity does not have any quantification over arguments. In [55], the authors propose a normal-form bisimilarity which decomposes values using *ultimate patterns* which describe the structure of a value along with some type information: values are then equated if their ultimate patterns can be related.

Environmental bisimulation has also been defined for a calculus with recursive and existential types in [85], a result extended to a store construct in [82]. Existential types restrict the way terms can be tested. Indeed, suppose we have two pairs $t_0 \stackrel{\mathsf{def}}{=} \langle 1, \lambda x^{\mathsf{int}}.x = 0\rangle$ and $t_1 \stackrel{\mathsf{def}}{=} \langle \mathsf{true}, \lambda x^{\mathsf{bool}}.\neg x\rangle$, both typed $\exists A.A \times (A \to \mathsf{bool})$. A context cannot distinguish t_0 and t_1, because the only term of type A a context has access to is 1 from t_0 and true from t_1, and when these values are passed to respectively $\lambda x^{\mathsf{int}}.x = 0$ and $\lambda x^{\mathsf{bool}}.\neg x$, we obtain the same result. However, the authors of [44] argue that applicative bisimilarity cannot relate these terms, because it tests λ-abstractions with arbitrary values. Therefore, more evolved relations are required, such as environmental bisimilarity. Indeed, environmental bisimilarity can store in its environment the values 1 and true with type A, and then use them to compare the λ-abstractions in t_0 and t_1.

Danvy and Filinski propose a simply typed calculus with shift and reset in [17], and parametric polymorphism has then been added to the language by Asai and Kamayema in [5]. In such systems, types are assigned not only to terms, but also to contexts. Pure contexts E are given types of the form $A \rhd B$, where A is the type of the hole and B is the answer type, and evaluation contexts (also called metacontexts) F are assigned types of the form $\neg A$, where A is the type of the hole. A typing judgment $\Gamma \mid B \vdash t : A \mid C$ roughly means that under the typing context Γ, the term t can be plugged into a pure context E of type $A \rhd B$ and a metacontext F of type $\neg C$, producing a well-typed term $F[\langle E[t]\rangle]$. In general, the evaluation of t may capture the surrounding context of type $A \rhd B$ to produce a value of type C, with $B \neq C$. Function types also contain extra information about the contexts the terms are plugged into: a term of type $A\ {}_C\!\to_D B$ can be applied to an argument of type A within a pure context of type $B \rhd C$ and a metacontext of type $\neg D$.

The complexity of the type systems for shift and reset (compared to, e.g., plain λ-calculus) may have some consequences on the definition of a typed bisimilarity for the language. In particular, we wonder how the extra type annotations for pure contexts and metacontexts should be factored in the bisimilarities. It seems natural to include types for the pure contexts for control stuck terms, since pure contexts already occur in the definitions of applicative and environmental bisimilarities in that case; it is not clear if and how the types for the metacontexts should be mentioned. The study of a typed $\lambda_{\mathcal{S}}$ can be interesting also to see how the types modify the equivalences between terms, in particular we wonder if the equivalences proved in this paper still hold in the presence of types. We leave this as a future work.

A related and unexplored topic is defining logical relations to characterize contextual equivalence for typed calculi with delimited continuations. So far, Asai introduced logical relation to prove the correctness of a partial evaluator for shift and reset [4], whereas Biernacka et al. in a series of articles proposed logical predicates for proving termination of evaluation in several calculi of delimited control [6, 8, 9]. We expect such logical relations to exploit the notion of context and metacontext and, therefore, to be biorthogonal [47, 67]. Biorthogonal and step-indexed Kripke logical relations have been proposed for an ML-like language with $\mathsf{call/cc}$ by Dreyer et al. [21] and adapting this approach to a similar language based on Asai and Kameyama's polymorphic type system for shift and reset [5] presents itself as an interesting topic of future research. An alternative to step-indexed Kripke logical relations that also have been shown to account for abortive continuations are parametric bisimulations [36], built on relation transition systems of Hur et al. [35]. Whether such hybrids of logical relations and bisimulations can effectively support reasoning about delimited continuations is an open question.

7.6. Other Delimited-Control Operators

CPS hierarchy. The operator shift and reset are just an instance of a more general construct called the *CPS hierarchy* [18]. As explained in Section 2.4, shift and reset have been originally defined by a translation into CPS. When iterated, the CPS translation leads to a hierarchy of continuations, in which it is possible to define a hierarchy of control operators shift_i and reset_i ($i \geq 1$) that generalize shift and reset, and that make it possible to separate computational effects that should exist independently in a program. For example, in order to collect the solutions found by a backtracking algorithm implemented with shift_1 and reset_1, one has to employ shift_2 and reset_2, so that there is no interference between searching and emitting the results of the search. The CPS hierarchy was also envisaged to account for nested computations in hierarchical structures. Indeed, as shown in [7], the hierarchy naturally accounts for normalization by evaluation algorithms for hierarchical languages of units and products, generalizing the problem of computing disjunctive or conjunctive normal forms in propositional logic.

In the hierarchy, a shift operator of level i captures the context up to the first enclosing reset_i or reset_j with $j > i$. So for example in $\langle E_1[\langle E_0[\mathcal{S}_1 k.t]\rangle_2]\rangle_1$, the $\mathcal{S}_1$ captures only E_0, not E_1. We believe the results of this paper generalize to the CPS hierarchy without issues. The notions of pure context and control stuck term now depend on the hierarchy level: a pure context of level i does not contain a reset_j (for $j \geq i$) encompassing its hole, and can be captured by an operator shift_i. A control stuck term of level i is an operator shift_i in a pure context of level i. The definitions of bisimulations have to be generalized to deal with control stuck terms of level i the same way we treat stuck terms of level 1. For example, two control stuck terms of level i are applicative bisimilar if they are bisimilar when put in an arbitrary level i pure context surrounded by a reset_i. The proofs for $i = 1$ should carry through to any i.

shift_0. The operator shift_0 ($\mathcal{S}_0$) allows a term to capture a pure context with its enclosing delimiter [18]. The capture reduction rule for this operator is thus as follows:

$$F[\langle E[\mathcal{S}_0 k.t]\rangle] \rightarrow_{\mathrm{v}} F[t\{\lambda x.\langle E[x]\rangle/k\}], \text{ with } x \notin \mathsf{fv}(E)$$

Note that there is no reset around t in $F[t\{\lambda x.\langle E[x]\rangle/k\}]$. Consequently, a term is able to directly decompose an evaluation context F into pure contexts through successive captures with $\mathcal{S}_0$; this is not possible in $\lambda_{\mathcal{S}}$.

The definitions of bisimilarities of this paper should extend to a calculus with shift_0 as far as the relaxed semantics is concerned. Since a term is able to access the context beyond the first enclosing reset, contextual equivalence is more discriminating with shift_0 than in $\lambda_{\mathcal{S}}$. For example, $\langle\langle t\rangle\rangle$ is no longer equivalent to $\langle t\rangle$, as we can see by taking $t = \mathcal{S}_0 k.\mathcal{S}_0 k.\Omega$.

For the original semantics (that in the case of shift_0 assumes a persistent top-level reset), the definitions have to take into account the fact that a delimited term $\langle t\rangle$ may evaluate to a control stuck term (like, e.g., $\langle \mathcal{S}_0 k.\mathcal{S}_0 k.t\rangle$ for any t) and that, therefore, it is not sufficient to compare values with values only. For instance, in order to validate the following equation taken from the axiomatization of shift_0 [59]:

$$\mathcal{S}_0 k.\langle(\lambda x.\mathcal{S}_0 k'.k\ x)\ t\rangle =_{\mathrm{M}} t, \text{ with } k \notin \mathsf{fv}(t)$$

we would have to be able to compare normal forms of different kinds, which can be achieved by putting the normal forms in a context $\langle E\rangle$ for any E.

$\mathsf{control}$ *and* prompt. The $\mathsf{control}$ operator ($\mathcal{F}$) captures a pure context up to the first enclosing prompt ($\#$), but the captured context does not include the delimiter [24]. Formally, the capture reduction rule is as follows:

$$F[\# E[\mathcal{F} k.t]] \rightarrow_{\mathrm{v}} F[\# t\{\lambda x.E[x]/k\}], \text{ with } x \notin \mathsf{fv}(E)$$

Unlike with shift and reset where continuation composition is static, with $\mathsf{control}$ and prompt it is dynamic, in the sense that the extent of control operations in the captured context comprises the context of the resumption of the captured context [10]. A $\mathsf{control}_0$ variant also exists [78], where the delimiter is captured with the context but not kept: as a result, no delimiter is present in the right-hand side of the capture reduction rule.

The theory of this paper should extend to $\mathsf{control}$ and prompt with minor changes. However, studying this calculus would still be interesting to pinpoint the differences between the equivalences of $\mathsf{shift}/\mathsf{reset}$ and $\mathsf{control}/\mathsf{prompt}$. For example, $\#\# t$ is equivalent to $\# t$, the same way $\langle\langle t\rangle\rangle$ is equivalent to $\langle t\rangle$. In fact, we conjecture the axioms can still be proved equivalent if we replace shift and reset with $\mathsf{control}$ and prompt (with the same restriction for $\mathcal{S}_{\mathtt{elim}}$). In contrast, $t_0 \overset{\mathsf{def}}{=} (\mathcal{S} k_1.k_1\ (\lambda x.\mathcal{S} k_2.t)\ \Omega)\ v$ (where $k_1, k_2, x \notin \mathsf{fv}(t)$) is equivalent to $\mathcal{S} k.\Omega$ (because $t_0 \xrightarrow{E}\rightarrow_{\mathrm{v}}^* \langle\langle E[\mathcal{S} k_2.t]\rangle\ \Omega\rangle$, and this term always diverges), but the term $t_0' \overset{\mathsf{def}}{=} (\mathcal{F} k_1.k_1\ (\lambda x.\mathcal{F} k_2.t)\ \Omega)\ v$ is equivalent to $\# t$ (because $t_0' \xrightarrow{E}\rightarrow_{\mathrm{v}}^* \# E[\mathcal{S} k_2.t]\ \Omega \rightarrow_{\mathrm{v}} \# t$). Maybe we can find (general enough) laws which hold with $\mathsf{control}$ and prompt but not with shift and reset, and conversely.

Multiple prompts. In languages with (named) multiple prompts [31, 23, 20] control delimiters (prompts) as well as control operators are tagged with names, so that the control operator captures the evaluation context up to the dynamically nearest delimiter with the matching name. In a calculus with tagged shift ($\mathcal{S}_a$) and reset ($\langle\cdot\rangle_a$) the operational semantics of shift is given by the following rule:

$$F'[\langle F[\mathcal{S}_a k.t]\rangle_a] \rightarrow_{\mathrm{v}} F'[\langle t\{\lambda x.\langle F[x]\rangle_a/k\}\rangle_a], \text{ with } a \notin \#(F) \text{ and } x \notin \mathsf{fv}(E)$$

where $\#(F)$ is the set of the prompts guarding the hole of F. Such calculi resemble the CPS hierarchy, already considered in this section, however there are differences in their semantics. In contrast to the CPS hierarchy, where evaluation contexts form a hierarchy[3] and the extent of control operations of level i is limited by control delimiters of any level $j \geq i$, in the calculus with multiple prompts the evaluation context is a list of the standard CBV evaluation contexts separated by named prompts and the control operations reach across any prompts up to a matching one. Moreover, the salient and unique feature of such calculi is dynamic name generation that allows one, e.g., to eliminate unwanted interactions between the control operations used to implement some control structure (e.g., coroutines) and the control operations of the code that uses the control structure.

Even without dynamic name generation, which gives an additional expressive power to such calculi, calculi with multiple prompts generalize, e.g., simple exceptions [31] and the catch/throw constructs [15]. The results of this article can be seamlessly adapted to these calculi and most, if not all, of the presented techniques should carry over without surprises.

However, when dynamic name generation is included in the calculus, comparing two terms becomes more difficult, as prompts with the same purpose can be generated with different names. With environmental bisimilarity, we can use environments to remember the relationships between generated prompts. We do so in [3] and define sound and complete environmental bisimilarities and their up-to techniques for a calculus with dynamically generated prompt names. Resource generation makes the definition of a sound applicative bisimilarity difficult for such a calculus, as argued in [44].

7.7. Other Constructs

Here, we briefly discuss what happens when $\lambda_{\mathcal{S}}$ is extended with constructs that can be found in usual programming languages.

Constants. While adding constants (such as numerals, booleans, ...) to the language does not raise any issue for applicative [29] and environmental bisimilarities, defining a satisfactory normal-form bisimilarity in the presence of constants raises some difficulties. On one hand, we do not have to tweak the

[3] In the original semantics, the evaluation context of level $i+1$ is a list (a stack, really) of evaluation contexts of level i separated by control delimiters of level i (contexts of level 1 are just the standard CBV evaluation contexts.) and the number of context layers is fixed [7].

$$
\begin{array}{rl}
\text{relaxed semantics:} & \mathbb{N} \subsetneq \mathbb{R} \subsetneq \mathbb{C} = \mathbb{A} = \mathbb{E} \\
 & \phantom{\equiv \subsetneq \mathbb{M} \subsetneq}\ \rotatebox{-90}{$\subsetneq$} \\
\text{original semantics:} & \equiv\ \subsetneq \mathbb{M} \subsetneq \mathbb{P} = \mathbb{F}
\end{array}
$$

Figure 8: Relationships between the equivalences of $\lambda_{\mathcal{S}}$

definition of normal-form bisimilarity to validate the η law (with the $\star$ operator), as it does no longer hold with constants. However, as pointed out in [80], the difficulty is in relating open terms featuring constant expressions: e.g., it is not clear how to define a normal-form bisimulation which equates $x+y$ and $y+x$. Relying on encodings of constants into plain λ-calculus is not enough, as these encodings usually do not respect the properties of the constants, like, for example, commutativity of $+$.

Store. Bisimilarities for languages with store are usually of the environmental kind [74, 46, 82], and [44] argues that the usual form of applicative bisimilarity is not sound in presence of store. An exception is Støvring and Lassen's study of $\lambda\mu\rho$ [80] (a calculus with store and an abortive control construct inspired by Parigot's $\lambda\mu$ [65]), where normal-form bisimilarity is sound and complete. But the work in [80] relies on the fact that in $\lambda\mu$-calculus (and in $\lambda\mu\rho$ as well), terms are of the form $[a]t$, where the name a acts as a placeholder for an evaluation context. These names are also essential to be able to define a sound and complete applicative bisimilarity for $\lambda\mu$ [14]. We do not have such names in $\lambda_{\mathcal{S}}$, so it is not clear that adding store to the calculus would lead to a sound and complete normal-form bisimilarity as in $\lambda\mu\rho$. We leave this study as a future work.

Exceptions. Like for store, Koutavas et al. give examples in [44] showing that applicative bisimilarity is not sound for a calculus with exceptions, and environmental bisimilarity should instead be used. Studying an extension of $\lambda_{\mathcal{S}}$ with exceptions would be interesting to compare the encoding of exceptions using shift and reset [26] to the native constructs. We leave this as a future work.

8. Conclusion and Future Work

In our study of the behavioral theory of a calculus with shift and reset, we consider two semantics: the original one, where terms are executed within an outermost reset, and the relaxed one, where this requirement is lifted. For each, we define a contextual equivalence (respectively $\mathbb{P}$ and $\mathbb{C}$), that we try to characterize with different kinds of bisimilarities (normal-form $\mathbb{N}$, $\mathbb{R}$, $\mathbb{M}$, applicative $\mathbb{A}$, and environmental $\mathbb{E}$, $\mathbb{F}$). We also compare our relations to CPS equivalence $\equiv$, a relation which equates terms with $\beta\eta$-equivalent CPS translations. We summarize in Figure 8 the relationships we proved in this paper between these relations.

When comparing term equivalence proofs, we can see that each bisimulation style has its strengths and weaknesses. Normal-form bisimulation arguably leads to the simplest proofs of equivalence in most cases, as it does not contain any quantification over arguments or testing contexts in its definition. For example, the β_Ω axiom can be easily proved using normal-form bisimulation (Proposition 4.31); the proof with applicative or environmental bisimulations are much more complex (Propositions 5.23 and 6.26). Normal-form bisimulation can also be combined with up-to techniques, leading to minimal proof obligations. Indeed, proving that Turing's fixed-point combinator is bisimilar to its shift/reset variant can be done with a normal-form bisimulation up to context with two pairs only (Example 4.10). Proving the same result with applicative bisimulation requires more work (see Example 5.9), since the definition of up-to techniques remains an open problem for that style. With environmental bisimulation up to context, we also need only two pairs of terms (see Example 6.22), but one has to find the (not completely obvious) environment under which to compare them.

However, normal-form bisimulation cannot be used to prove all equivalences, since its corresponding bisimilarity is not complete. It can be too discriminating to relate very simple terms, like those in Propositions 4.18 and 4.7, even though refined normal-form bisimulation (Section 4.4) can help. Besides, normal-form bisimulation operates on open terms by definition, which requires to consider an extra normal form (open stuck terms) in the bisimulation proofs. Applicative and environmental bisimulations do not have these issues: their corresponding bisimilarities are complete, and they operate on closed terms. As a result, the proof that $\langle\langle t\rangle\rangle$ is equivalent to $\langle t\rangle$ is shorter with applicative bisimulation than with normal-form bisimulation (compare Example 4.3 and Example 5.8), and even shorter if we use environmental bisimulation up to context (see Example 6.21). This is also true, e.g., for the $\langle\cdot\rangle_{\mathtt{lift}}$ axiom (compare Proposition 4.30, 5.22, and 6.25).

To summarize, to prove that two given terms are equivalent, we would suggest to first try normal-form bisimulation and its up-to techniques, and if it fails, try applicative bisimulation, and next, environmental bisimulation (with its up-to techniques). The relations defined for the relaxed semantics can also be used as equivalence proof techniques for the original semantics, except if one wants to relate a control stuck term with a term which does not necessarily evaluate to a control stuck term, like, e.g., in the $\mathcal{S}_{\mathtt{elim}}$ axiom. In that case, only the normal-form and environmental bisimulations dedicated to the original semantics can be used.

Future work. Among the extensions discussed in Section 7, we believe that CPS-based behavioral equivalences (Section 7.3), the $\lambda\mu\widehat{\mathsf{tp}}$ calculus (Section 7.4), typed setting (Section 7.5), control and prompt (Section 7.6), and store (Section 7.7) are worth exploring.

The behavioral theory of abortive control operators is also interesting to look at, as it is more challenging than for delimited-control operators. Although our results for shift and reset carry over to delimited versions of abortive control operators call/cc and abort, where the computation can be aborted up to the nearest enclosing control delimiter [26, 38], they do not cater for the undelimited abortive control operators such as, e.g., call/cc in Scheme, even if we restrict ourselves to an implicit top-level reset. First, our notions of program equivalence are based on multi-layered contexts, which can be too discriminating for one-layer aware abortive operators. Second, with an abortive control operator reduction is not preserved by evaluation contexts: $t \longrightarrow t'$ does not necessarily imply $E[t] \longrightarrow E[t']$ (e.g., if a capture happens in the first reduction), whereas the properties of the calculus we consider critically depend on the compatibility of the reduction relation.

The situation is a bit better in the $\lambda\mu$-calculus [65], because names allow us to see if a context is captured or not. As a result, some work has been carried out for variants of the $\lambda\mu$-calculus [50, 54, 80, 14]. But as far as we know, the behavioral theory of a language similar to the λ-calculus extended with call/cc has not been established.

[1] S. Abramsky and C.-H. L. Ong. Full abstraction in the lazy lambda calculus. *Information and Computation*, 105:159–267, 1993.

[2] Z. M. Ariola, H. Herbelin, and A. Sabry. A proof-theoretic foundation of abortive continuations. *Higher-Order and Symbolic Computation*, 20(4):403–429, 2007.

[3] A. Aristizábal, D. Biernacki, S. Lenglet, and P. Polesiuk. Environmental Bisimulations for Delimited-Control Operators with Dynamic Prompt Generation. In D. Kesner and B. Pientka, editors, *1st International Conference on Formal Structures for Computation and Deduction (FSCD 2016)*, volume 52 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 9:1–9:17, Dagstuhl, Germany, 2016. Schloss Dagstuhl–Leibniz-Zentrum für Informatik.

[4] K. Asai. Logical relations for call-by-value delimited continuations. In M. van Eekelen, editor, *Proceedings of the 6th Symposium Trends in Functional Programming (TFP'05)*, pages 413–428, Tallinn, Estonia, Sept. 2005.

[5] K. Asai and Y. Kameyama. Polymorphic delimited continuations. In Z. Shao, editor, *Proceedings of the 5th Asian Symposium on Programming Languages and Systems (APLAS'07)*, number 4807 in LNCS, pages 239–254, Singapore, Dec. 2007.

[6] M. Biernacka and D. Biernacki. Context-based proofs of termination for typed delimited-control operators. In F. J. López-Fraguas, editor, *Proceedings of the 11th International Conference on Principles and Practice of Declarative Programming (PPDP'09)*, Coimbra, Portugal, Sept. 2009.

[7] M. Biernacka, D. Biernacki, and O. Danvy. An operational foundation for delimited continuations in the CPS hierarchy. *Logical Methods in Computer Science*, 1(2:5):1–39, Nov. 2005.

[8] M. Biernacka, D. Biernacki, and S. Lenglet. Typing control operators in the CPS hierarchy. In M. Hanus, editor, *Proceedings of the 13th International Conference on Principles and Practice of Declarative Programming (PPDP'11)*, Odense, Denmark, July 2011.

[9] M. Biernacka, D. Biernacki, S. Lenglet, and M. Materzok. Proving termination of evaluation for system F with control operators. In U. de'Liguoro and A. Saurin, editors, *Proceedings of the 1st Workshop on Control Operators and their Semantics (COS 2013)*, volume 127 of *EPTCS*, pages 15–27, Eindhoven, The Netherlands, June 2013.

[10] D. Biernacki, O. Danvy, and C. Shan. On the static and dynamic extents of delimited continuations. *Science of Computer Programming*, 60(3):274–297, 2006.

[11] D. Biernacki and S. Lenglet. Applicative bisimulations for delimited-control operators. In L. Birkedal, editor, *Foundations of Software Science and Computation Structures, 15th International Conference (FOSSACS'12)*, number 7213 in LNCS, pages 119–134, Tallinn, Estonia, Mar. 2012.

[12] D. Biernacki and S. Lenglet. Normal form bisimulations for delimited-control operators. In T. Schrijvers and P. Thiemann, editors, *Functional and Logic Programming, 13th International Symposium (FLOPS'12)*, number 7294 in LNCS, pages 47–61, Kobe, Japan, May 2012.

[13] D. Biernacki and S. Lenglet. Environmental bisimulations for delimited-control operators. In C. Shan, editor, *Proceedings of the 11th Asian Symposium on Programming Languages and Systems (APLAS'13)*, volume 8301 of *LNCS*, pages 333–348, Melbourne, VIC, Australia, Dec. 2013.

[14] D. Biernacki and S. Lenglet. Applicative bisimilarities for call-by-name and call-by-value $\lambda\mu$-calculus. In B. Jacobs, A. Silva, and S. Staton, editors, *Proceedings of the 30th Annual Conference on Mathematical Foundations of Programming Semantics(MFPS XXX)*, volume 308 of *ENTCS*, pages 49–64, Ithaca, NY, USA, June 2014.

[15] T. Crolard. A confluent lambda-calculus with a catch/throw mechanism. *Journal of Functional Programming*, 9(6):625–647, 1999.

[16] O. Danvy. Type-directed partial evaluation. In Steele Jr. [79], pages 242–257.

[17] O. Danvy and A. Filinski. A functional abstraction of typed contexts. DIKU Rapport 89/12, DIKU, Computer Science Department, University of Copenhagen, Copenhagen, Denmark, July 1989.

[18] O. Danvy and A. Filinski. Abstracting control. In M. Wand, editor, *Proceedings of the 1990 ACM Conference on Lisp and Functional Programming*, pages 151–160, Nice, France, June 1990.

[19] O. Danvy and A. Filinski. Representing control, a study of the CPS transformation. *Mathematical Structures in Computer Science*, 2(4):361–391, 1992.

[20] P. Downen and Z. M. Ariola. A systematic approach to delimited control with multiple prompts. In H. Seidl, editor, *21st European Symposium on Programming (ESOP'12)*, volume 7211 of *LNCS*, pages 234–253, Tallinn, Estonia, Mar. 2012.

[21] D. Dreyer, G. Neis, and L. Birkedal. The impact of higher-order state and control effects on local relational reasoning. *Journal of Functional Programming*, 22(4-5):477–528, 2012.

[22] P. Dybjer and A. Filinski. Normalization and partial evaluation. In G. Barthe, P. Dybjer, L. Pinto, and J. Saraiva, editors, *Applied Semantics – Advanced Lectures*, number 2395 in LNCS, pages 137–192, Caminha, Portugal, Sept. 2000.

[23] R. K. Dybvig, S. Peyton-Jones, and A. Sabry. A monadic framework for delimited continuations. *Journal of Functional Programming*, 17(6):687–730, 2007.

[24] M. Felleisen. The theory and practice of first-class prompts. In J. Ferrante and P. Mager, editors, *Proceedings of the Fifteenth Annual ACM Symposium on Principles of Programming Languages*, pages 180–190, San Diego, California, Jan. 1988.

[25] M. Felleisen and R. Hieb. The revised report on the syntactic theories of sequential control and state. *Theoretical Computer Science*, 103(2):235–271, 1992.

[26] A. Filinski. Representing monads. In H.-J. Boehm, editor, *Proceedings of the Twenty-First Annual ACM Symposium on Principles of Programming Languages*, pages 446–457, Portland, Oregon, Jan. 1994.

[27] A. Filinski. Representing layered monads. In A. Aiken, editor, *Proceedings of the Twenty-Sixth Annual ACM Symposium on Principles of Programming Languages*, pages 175–188, San Antonio, Texas, Jan. 1999.

[28] M. Flatt, G. Yu, R. B. Findler, and M. Felleisen. Adding delimited and composable control to a production programming environment. In N. Ramsey, editor, *Proceedings of the 2007 ACM SIGPLAN International Conference on Functional Programming (ICFP'07)*, pages 165–176, Freiburg, Germany, Sept. 2007.

[29] A. D. Gordon. Bisimilarity as a theory of functional programming. *Theoretical Computer Science*, 228(1-2):5–47, 1999.

[30] A. D. Gordon and G. D. Rees. Bisimilarity for a first-order calculus of objects with subtyping. In Steele Jr. [79], pages 386–395.

[31] C. Gunter, D. Rémy, and J. G. Riecke. A generalization of exceptions and control in ML-like languages. In S. Peyton Jones, editor, *Proceedings of the Seventh ACM Conference on Functional Programming and Computer Architecture*, pages 12–23, La Jolla, California, June 1995.

[32] H. Herbelin and S. Ghilezan. An approach to call-by-name delimited continuations. In P. Wadler, editor, *Proceedings of the Thirty-Fifth Annual ACM Symposium on Principles of Programming Languages*, pages 383–394, San Francisco, CA, USA, Jan. 2008.

[33] R. Hieb, R. K. Dybvig, and C. W. Anderson, III. Subcontinuations. *Lisp and Symbolic Computation*, 5(4):295–326, Dec. 1993.

[34] D. J. Howe. Proving congruence of bisimulation in functional programming languages. *Information and Computation*, 124(2):103–112, 1996.

[35] C. Hur, D. Dreyer, G. Neis, and V. Vafeiadis. The marriage of bisimulations and Kripke logical relations. In J. Field and M. Hicks, editors, *Proceedings of the Thirty-Ninth Annual ACM Symposium on Principles of Programming Languages*, pages 59–72, Philadelphia, PA, USA, Jan. 2012.

[36] C.-K. Hur, G. Neis, D. Dreyer, and V. Vafeiadis. A logical step forward in parametric bisimulations. Technical Report MPI-SWS-2014-003, Max Planck Institute for Software Systems (MPI-SWS), Saarbrcken, Germany, Jan. 2014.

[37] Y. Kameyama. Axioms for control operators in the CPS hierarchy. *Higher-Order and Symbolic Computation*, 20(4):339–369, 2007.

[38] Y. Kameyama and M. Hasegawa. A sound and complete axiomatization of delimited continuations. In O. Shivers, editor, *Proceedings of the 2003 ACM SIGPLAN International Conference on Functional Programming (ICFP'03)*, pages 177–188, Uppsala, Sweden, Aug. 2003.

[39] Y. Kameyama and A. Tanaka. Equational axiomatization of call-by-name delimited control. In T. Kutsia, W. Schreiner, and M. Fernández, editors, *Proceedings of the 12th International ACM SIGPLAN Conference on Principles and Practice of Declarative Programming*, pages 77–86, Hagenberg, Austria, July 2010.

[40] O. Kiselyov. Delimited control in OCaml, abstractly and concretely: System description. In M. Blume and G. Vidal, editors, *Functional and Logic Programming, 10th International Symposium, FLOPS 2010*, number 6009 in LNCS, pages 304–320, Sendai, Japan, Apr. 2010.

[41] O. Kiselyov and C. Shan. Delimited continuations in operating systems. In B. Kokinov, D. C. Richardson, T. R. Roth-Berghofer, and L. Vieu, editors, *Modeling and Using Context, 6th International and Interdisciplinary Conference, CONTEXT 2007*, number 4635 in LNAI, pages 291–302, Roskilde, Denmark, Aug. 2007.

[42] O. Kiselyov and C. Shan. Embedded probabilistic programming. In W. Taha, editor, *Domain-Specific Languages, DSL 2009*, number 5658 in LNCS, pages 360–384, Oxford, UK, July 2009.

[43] O. Kiselyov, C. Shan, D. P. Friedman, and A. Sabry. Backtracking, interleaving, and terminating monad transformers. In B. Pierce, editor, *Proceedings of the 2005 ACM SIGPLAN International Conference on Functional Programming (ICFP'05)*, pages 192–203, Tallinn, Estonia, Sept. 2005.

[44] V. Koutavas, P. B. Levy, and E. Sumii. From applicative to environmental bisimulation. In M. Mislove and J. Ouaknine, editors, *Proceedings of the 27th Annual Conference on Mathematical Foundations of Programming Semantics(MFPS XXVII)*, volume 276 of *ENTCS*, pages 215–235, Pittsburgh, PA, USA, May 2011.

[45] V. Koutavas and M. Wand. Bisimulations for untyped imperative objects. In P. Sestoft, editor, *15th European Symposium on Programming, ESOP 2006*, volume 3924 of *LNCS*, pages 146–161, Vienna, Austria, Mar. 2006.

[46] V. Koutavas and M. Wand. Small bisimulations for reasoning about higher-order imperative programs. In J. G. Morrisett and S. L. P. Jones, editors, *Proceedings of the 33rd Annual ACM Symposium on Principles of Programming Languages*, pages 141–152, Charleston, SC, USA, Jan. 2006.

[47] J. Krivine. Classical logic, storage operators and second-order lambda-calculus. *Annals of Pure and Applied Logic*, 68(1):53–78, 1994.

[48] I. Lanese, J. A. Pérez, D. Sangiorgi, and A. Schmitt. On the expressiveness and decidability of higher-order process calculi. *Information and Computation*, 209(2):198–226, 2011.

[49] S. B. Lassen. Relational reasoning about contexts. In A. D. Gordon and A. M. Pitts, editors, *Higher Order Operational Techniques in Semantics*. Cambridge University Press, 1998. 91-135.

[50] S. B. Lassen. Bisimulation for pure untyped $\lambda\mu$-caluclus (extended abstract). Unpublished note, Jan. 1999.

[51] S. B. Lassen. Bisimulation in untyped lambda calculus: Böhm trees and bisimulation up to context. In M. M. Stephen Brookes, Achim Jung and A. Scedrov, editors, *Proceedings of the 15th Annual Conference on Mathematical Foundations of Programming Semantics*, volume 20 of *ENTCS*, pages 346–374, New Orleans, LA, Apr. 1999.

[52] S. B. Lassen. Eager normal form bisimulation. In P. Panangaden, editor, *Proceedings of the 20th IEEE Symposium on Logic in Computer Science (LICS 2005)*, pages 345–354, Chicago, IL, June 2005.

[53] S. B. Lassen. Normal form simulation for McCarthy's amb. In M. Escardó, A. Jung, and M. Mislove, editors, *Proceedings of the 21st Annual Conference on Mathematical Foundations of Programming Semantics(MFPS XXI)*, volume 155 of *ENTCS*, pages 445–465, Birmingham, UK, May 2005.

[54] S. B. Lassen. Head normal form bisimulation for pairs and the $\lambda\mu$-calculus. In R. Alur, editor, *Proceedings of the 21st Annual IEEE Symposium on Logic in Computer Science (LICS 2006)*, pages 297–306, Seattle, WA, USA, Aug. 2006.

[55] S. B. Lassen and P. B. Levy. Typed normal form bisimulation. In J. Duparc and T. A. Henzinger, editors, *Computer Science Logic, CSL'07*, volume 4646 of *LNCS*, pages 283–297, Lausanne, Switzerland, Sept. 2007.

[56] S. B. Lassen and P. B. Levy. Typed normal form bisimulation for parametric polymorphism. In F. Pfenning, editor, *Proceedings of the 23rd IEEE Symposium on Logic in Computer Science (LICS 2008)*, pages 341–352, Pittsburgh, PA, USA, June 2008.

[57] S. Lenglet, A. Schmitt, and J.-B. Stefani. Howe's method for calculi with passivation. In M. Bravetti and G. Zavattaro, editors, *The 20th International Conference on Concurrency Theory*, number 5710 in LNCS, pages 448–462, Bologna, Italy, July 2009.

[58] J.-M. Madiot. *Higher-order languages: dualities and bisimulation enhancements*. PhD thesis, Université de Lyon and Università di Bologna, 2015.

[59] M. Materzok. Axiomatizing subtyped delimited continuations. In S. R. D. Rocca, editor, *Computer Science Logic 2013, CSL 2013*, volume 23 of *LIPIcs*, pages 521–539, Torino, Italy, Sept. 2013.

[60] M. Materzok and D. Biernacki. Subtyping delimited continuations. In O. Danvy, editor, *Proceedings of the 2011 ACM SIGPLAN International Conference on Functional Programming (ICFP'11)*, pages 81–93, Tokyo, Japan, Sept. 2011.

[61] M. Merro and C. Biasi. On the observational theory of the CPS-calculus: (extended abstract). In S. Brookes and M. Mislove, editors, *Proceedings of the 22nd Annual Conference on Mathematical Foundations of Programming Semantics(MFPS XXII)*, volume 158 of *ENTCS*, pages 307–330, Genova, Italy, May 2006.

[62] R. Milner. Fully abstract models of typed λ-calculi. *Theoretical Computer Science*, 4(1):1–22, 1977.

[63] L. Moreau and C. Queinnec. Partial continuations as the difference of continuations, a duumvirate of control operators. In M. Hermenegildo and J. Penjam, editors, *Sixth International Symposium on Programming Language Implementation and Logic Programming*, number 844 in LNCS, pages 182–197, Madrid, Spain, Sept. 1994.

[64] J. H. Morris. *Lambda Calculus Models of Programming Languages.* PhD thesis, Massachusets Institute of Technology, 1968.

[65] M. Parigot. $\lambda\mu$-calculus: an algorithmic interpretation of classical natural deduction. In A. Voronkov, editor, *Proceedings of the International Conference on Logic Programming and Automated Reasoning (LPAR'92)*, number 624 in LNAI, pages 190–201, St. Petersburg, Russia, July 1992.

[66] A. Piérard and E. Sumii. A higher-order distributed calculus with name creation. In *Proceedings of the 27th IEEE Symposium on Logic in Computer Science (LICS 2012)*, pages 531–540, Dubrovnik, Croatia, June 2012.

[67] A. Pitts and I. Stark. Operational reasoning for functions with local state. In A. Gordon and A. Pitts, editors, *Higher Order Operational Techniques in Semantics*, pages 227–273. Publications of the Newton Institute, Cambridge University Press, 1998.

[68] A. M. Pitts. Parametric polymorphism and operational equivalence. *Mathematical Structures in Computer Science*, 10(3):321–359, 2000.

[69] G. D. Plotkin. Call-by-name, call-by-value and the λ-calculus. *Theoretical Computer Science*, 1:125–159, 1975.

[70] T. Rompf, I. Maier, and M. Odersky. Implementing first-class polymorphic delimited continuations by a type-directed selective cps-transform. In A. Tolmach, editor, *Proceedings of the 2009 ACM SIGPLAN International Conference on Functional Programming (ICFP'09)*, pages 317–328, Edinburgh, UK, Aug. 2009.

[71] A. Sabry. Note on axiomatizing the semantics of control operators. Technical Report CIS-TR-96-03, Department of Computer and Information Science, University of Oregon, 1996.

[72] D. Sangiorgi. The lazy lambda calculus in a concurrency scenario. In A. Scedrov, editor, *Proceedings of the Seventh Annual IEEE Symposium on Logic in Computer Science (LICS'92)*, pages 102–109, Santa Cruz, California, June 1992.

[73] D. Sangiorgi, N. Kobayashi, and E. Sumii. Environmental bisimulations for higher-order languages. In J. Marcinkowski, editor, *Proceedings of the 22nd IEEE Symposium on Logic in Computer Science (LICS 2007)*, pages 293–302, Wroclaw, Poland, July 2007.

[74] D. Sangiorgi, N. Kobayashi, and E. Sumii. Environmental bisimulations for higher-order languages. *ACM Transactions on Programming Languages and Systems*, 33(1):1–69, Jan. 2011.

[75] D. Sangiorgi and D. Walker. *The Pi-Calculus: A Theory of Mobile Processes.* Cambridge University Press, 2001.

[76] N. Sato and E. Sumii. The higher-order, call-by-value applied Pi-calculus. In Z. Hu, editor, *Proceedings of the 7th Asian Symposium on Programming Languages and Systems (APLAS'09)*, volume 5904 of *LNCS*, pages 311–326, Seoul, Korea, Dec. 2009.

[77] C. Shan. Delimited continuations in natural language: quantification and polarity sensitivity. In H. Thielecke, editor, *Proceedings of the Fourth ACM SIGPLAN Workshop on Continuations (CW'04)*, Technical report CSR-04-1, Department of Computer Science, Queen Mary's College, pages 55–64, Venice, Italy, Jan. 2004.

[78] C. Shan. A static simulation of dynamic delimited control. *Higher-Order and Symbolic Computation*, 20(4):371–401, 2007.

[79] G. L. Steele Jr., editor. *Proceedings of the Twenty-Third Annual ACM Symposium on Principles of Programming Languages*, St. Petersburg Beach, Florida, Jan. 1996. ACM Press.

[80] K. Støvring and S. B. Lassen. A complete, co-inductive syntactic theory of sequential control and state. In M. Felleisen, editor, *Proceedings of the 34th Annual ACM Symposium on Principles of Programming Languages*, pages 161–172, Nice, France, Jan. 2007.

[81] E. Sumii. An implementation of transparent migration on standard Scheme. In M. Felleisen, editor, *Proceedings of the Workshop on Scheme and Functional Programming*, Technical Report 00-368, Rice University, pages 61–64, Montréal, Canada, Sept. 2000.

[82] E. Sumii. A complete characterization of observational equivalence in polymorphic *lambda*-calculus with general references. In E. Grädel and R. Kahle, editors, *Computer Science Logic, CSL'09*, volume 5771 of *LNCS*, pages 455–469, Coimbra, Portugal, Sept. 2009.

[83] E. Sumii. A bisimulation-like proof method for contextual properties in untyped lambda-calculus with references and deallocation. *Theoretical Computer Science*, 411(51-52):4358–4378, 2010.

[84] E. Sumii and B. C. Pierce. A bisimulation for dynamic sealing. *Theoretical Computer Science*, 375(1-3):169–192, 2007.

[85] E. Sumii and B. C. Pierce. A bisimulation for type abstraction and recursion. *Journal of the ACM*, 54(5), 2007.

[86] H. Thielecke. *Categorical Structure of Continuation Passing Style*. PhD thesis, University of Edinburgh, Edinburgh, Scotland, 1997. ECS-LFCS-97-376.

[87] J. Tiuryn and M. Wand. Untyped lambda-calculus with input-output. In H. Kirchner, editor, *21st Colloquium on Trees in Algebra and Programming (CAAP'96)*, volume 1059 of *LNCS*, pages 317–329, Linköping, Sweden, Apr. 1996.

A. Normal-Form Bisimilarity

A.1. Normal-form bisimilarity

For this proof, we use a (very limited) version of bisimulation up to reduction. Given a relation $\mathcal{R}$, we write $t_0 \mathrel{\mathcal{R}^{\searrow}} t_1$ if there exist t'_0, t'_1 such that $t_0 \rightarrow^*_{\mathrm{v}} t'_0$, $t_1 \rightarrow^*_{\mathrm{v}} t'_1$, and $t'_0 \mathrel{\mathcal{R}} t'_1$. We define $\mathcal{R}^{\mathsf{C}\searrow}$ the same way as $\mathcal{R}^{\mathsf{C}}$ except on contexts of the form $F[\langle E\rangle]$, where we define it as follows:

$$\frac{\langle E_0[x]\rangle \mathrel{\mathcal{R}} \langle E_1[x]\rangle \qquad F_0[x] \mathrel{\mathcal{R}^{\searrow}} F_1[x] \qquad x \text{ fresh}}{F_0[\langle E_0\rangle] \mathrel{\mathcal{R}^{\mathsf{C}\searrow}} F_1[\langle E_1\rangle]}$$

and we define $\mathcal{R}^{\mathsf{NF}\eta\searrow}$ as $\mathcal{R}^{\mathsf{NF}\eta}$ except it uses $\mathcal{R}^{\mathsf{C}\searrow}$ to relate contexts.

Definition A.1. A relation $\mathcal{R}$ on terms is a normal-form simulation up to reduction if $t_0 \mathrel{\mathcal{R}} t_1$ implies:

- if $t_0 \rightarrow_{\mathrm{v}} t'_0$, then there exists t'_1 such that $t_1 \rightarrow^*_{\mathrm{v}} t'_1$, and there exist t''_0, t''_1 such that $t'_0 \rightarrow^*_{\mathrm{v}} t''_0$, $t'_1 \rightarrow^*_{\mathrm{v}} t''_1$, and $t''_0 \mathrel{\mathcal{R}} t''_1$;
- if t_0 is a normal form, then there exists t'_1 such that $t_1 \Downarrow_{\mathrm{v}} t'_1$ and $t_0 \mathrel{\mathcal{R}^{\mathsf{NF}\eta\searrow}} t'_1$.

A relation $\mathcal{R}$ is a normal-form bisimulation up to reduction if both $\mathcal{R}$ and $\mathcal{R}^{-1}$ are normal-form simulations up to reduction.

Lemma A.2. *If $\mathcal{R}$ is a bisimulation up to reduction, then $\mathcal{R} \subseteq \mathbb{N}$.*

We write $\overrightarrow{F}$ for a sequence of contexts $F_1 \dots F_n$, $\overrightarrow{F}[t]$ for $F_1[\dots F_n[t]]$, $\emptyset$ for an empty sequence, and $\overrightarrow{F}[\,]$ for the context represented by the sequence. For a relation $\mathcal{R}$ on contexts, we write $\overrightarrow{F_0} \mathrel{\mathcal{R}} \overrightarrow{F_1}$ if $\overrightarrow{F_0} = F^1_0 \dots F^m_0$, $\overrightarrow{F_1} = F^1_1 \dots F^m_1$, and for each $1 \leq i \leq m$, $F^i_0 \mathrel{\mathcal{R}} F^i_1$. We define the relation $(\overrightarrow{F_0}, t_0) \mathrel{[\mathbb{N}]} (\overrightarrow{F_1}, t_1)$ as:

$$\frac{t_0 \mathrel{\mathbb{N}} t_1}{(\emptyset, t_0) \mathrel{[\mathbb{N}]} (\emptyset, t_1)} \qquad \frac{(\overrightarrow{F_0}, t_0) \mathrel{[\mathbb{N}]} (\overrightarrow{F_1}, t_1) \qquad F'_0 \mathrel{\mathbb{N}^{\mathsf{C}}} F'_1}{(F'_0\overrightarrow{F_0}, t_0) \mathrel{[\mathbb{N}]} (F'_1\overrightarrow{F_1}, t_1)}$$

$$\frac{(\overrightarrow{F_0}, t_0) \mathrel{[\mathbb{N}]} (\overrightarrow{F_1}, t_1) \qquad \overrightarrow{F_0}[t_0], \overrightarrow{F_1}[t_1] \text{ delimited} \qquad F'_0[x] \mathrel{\mathbb{N}} F'_1[t_1] \qquad x \text{ fresh}}{(F'_0\overrightarrow{F_0}, t_0) \mathrel{[\mathbb{N}]} (F'_1\overrightarrow{F_1}, t_1)}$$

We write $\overrightarrow{F_0} \mathrel{[\mathbb{N}]} \overrightarrow{F_1}$, if $(\overrightarrow{F_0}, x) \mathrel{[\mathbb{N}]} (\overrightarrow{F_1}, x)$ holds for a fresh x.

Lemma A.3. *If $\overrightarrow{F_0} \mathrel{[\mathbb{N}]} \overrightarrow{F_1}$ and $\overrightarrow{F_0}[\,]$ is pure, then all the contexts in $\overrightarrow{F_0}$ and $\overrightarrow{F_1}$ are pure and pairwise related by $\mathbb{N}^{\mathsf{NF}\eta}$.*

Proof. By induction on the derivation of $(\overrightarrow{F_0}, x) \mathrel{[\mathbb{N}]} (\overrightarrow{F_1}, x)$ (x fresh). The rule which requires delimited terms in its premise can never be applied.

We define $v_0 \langle \mathbb{N} \rangle v_1$ as

$$\frac{v_0 \ \mathbb{N}^{\mathsf{NF}\eta} \ v_1}{v_0 \ \langle \mathbb{N} \rangle \ v_1} \qquad \frac{\overrightarrow{E_0} \ \mathbb{N}^{\mathsf{C}} \ \overrightarrow{E_1} \qquad \langle E_0' \rangle \ \mathbb{N}^{\mathsf{C}} \ \langle E_1' \rangle \qquad x \text{ fresh}}{\lambda x.\langle E_0'[\overrightarrow{E_0}[x]] \rangle \ \langle \mathbb{N} \rangle \ \lambda x.\langle E_1'[\overrightarrow{E_1}[x]] \rangle}$$

We write $\overrightarrow{\sigma}$ for a sequence of substitutions $\sigma_1 \ldots \sigma_n$, where each σ_i is of the form $\{v_i/x_i\}$, such that:

1. $x_i \neq x_j$ for $i \neq j$,
2. if $i \leq j$ then $x_i \notin \mathsf{fv}(v_j)$.

We write $t\vec{\sigma}$ for $(\ldots(t\sigma_0)\sigma_1 \ldots)\sigma_n$. The set $\{x_1, \ldots, x_n\}$ is the domain of $\overrightarrow{\sigma}$ noted $\mathsf{dom}(\overrightarrow{\sigma})$. When $x \in \mathsf{dom}(\overrightarrow{\sigma})$ and, therefore, is equal to some x_i, we write $\overrightarrow{\sigma}(x)$ for the value v_i. We also write $\overrightarrow{\sigma}\overrightarrow{\sigma'}$ for the concatenation of two sequences of substitutions $\overrightarrow{\sigma}$ and $\overrightarrow{\sigma'}$, if the resulting sequence satisfies the two conditions. We will often implicitly use the following lemma.

Lemma A.4. *If $\overrightarrow{\sigma}(x) = v$, then $x\overrightarrow{\sigma} = v\overrightarrow{\sigma}$.*

Proof. Follows directly from the two conditions on sequences of substitutions.

We write $\overrightarrow{\sigma_0} \ \langle \mathbb{N} \rangle \ \overrightarrow{\sigma_1}$ if $\sigma_0 = \{v_1^0/x_1\} \ldots \{v_n^0/x_n\}$, $\sigma_1 = \{v_1^1/x_1\} \ldots \{v_n^1/x_n\}$, and for each i, $v_i^0 \langle \mathbb{N} \rangle v_i^1$.

We define

$$\mathcal{R} \stackrel{\mathsf{def}}{=} \{(\overrightarrow{F_0}[t_0]\overrightarrow{\sigma_0}, \overrightarrow{F_1}[t_1]\overrightarrow{\sigma_1}) \mid (\overrightarrow{F_0}, t_0) \ [\mathbb{N}] \ (\overrightarrow{F_1}, t_1), \overrightarrow{\sigma_0} \ \langle \mathbb{N} \rangle \ \overrightarrow{\sigma_1}\}$$

Lemma A.5. *If $v_0 \ \mathbb{N}^{\mathsf{NF}\eta} \ v_1$ and $\overrightarrow{\sigma_0} \ \langle \mathbb{N} \rangle \ \overrightarrow{\sigma_1}$, then $v_0\overrightarrow{\sigma_0} \ \mathcal{R}^{\mathsf{NF}\eta\searrow} \ v_1\overrightarrow{\sigma_1}$.*

Proof. Immediate by the definition of $\mathcal{R}$.

Lemma A.6. *If $(\overrightarrow{F_0}F_0', t_0) \ [\mathbb{N}] \ (\overrightarrow{F_1}F_1', t_1)$ with $F_0'[x] \ \mathbb{N} \ F_1'[x]$ for a fresh x and some delimited terms t_0, t_1, then for all $v_0 \ \mathbb{N}^{\mathsf{NF}\eta} \ v_1$, there exist t_0', t_1' such that $\overrightarrow{F_0}[F_0'[v_0]] \rightarrow_{\mathrm{v}}^{*} t_0'$, $\overrightarrow{F_1}[F_1'[v_1]] \rightarrow_{\mathrm{v}}^{*} t_1'$, and $t_0' \ \mathcal{R} \ t_1'$.*

Proof. By induction on the size of $\overrightarrow{F_0}$, $\overrightarrow{F_1}$. If $\overrightarrow{F_0} = \overrightarrow{F_1} = \emptyset$, then we have $(\emptyset, F_0'[x]) \ [\mathbb{N}] \ (\emptyset, F_1'[x])$, which implies $F_0'[x]\{v_0/x\} \ \mathcal{R} \ F_1'[x]\{v_1/x\}$, hence the result holds.

Suppose $\overrightarrow{F_0} = \overrightarrow{F_0^{(3)}}F_0''$, $\overrightarrow{F_1} = \overrightarrow{F_1^{(3)}}F_1''$. If $F_0'' \ \mathbb{N}^{\mathsf{C}} \ F_1''$, then $(\overrightarrow{F_0}, F_0'[x]) \ [\mathbb{N}] \ (\overrightarrow{F_1}, F_1'[x])$, which implies $\overrightarrow{F_0}[F_0'[x]]\{v_0/x\} \ \mathcal{R} \ \overrightarrow{F_1}[F_1'[x]]\{v_1/x\}$, hence the result holds. Otherwise, we have $F_0''[y] \ \mathbb{N} \ F_1''[y]$ for a fresh y, and $F_0'[t_0]$, $F_1'[t_1]$ are delimited, which is possible only if either F_0' and F_1' are both delimited, or $F_0' = F_1' = \Box$, or if one is delimited and the other is $\Box$. In the first case, we have $(\overrightarrow{F_0}, F_0'[x]) \ [\mathbb{N}] \ (\overrightarrow{F_1}, F_1'[x])$, which implies $\overrightarrow{F_0}[F_0'[x]]\{v_0/x\} \ \mathcal{R} \ \overrightarrow{F_1}[F_1'[x]]\{v_1/x\}$. If $F_0' = F_1' = \emptyset$, then we have $(\overrightarrow{F_0^{(3)}}F_0'', t_0) \ [\mathbb{N}] \ (\overrightarrow{F_1^{(3)}}F_1'', t_1)$, and the induction hypothesis gives us the required result.

In the last case, we assume (without loss of generality) that $F'_0 = \Box$ and F'_1 is delimited. Since we have $F'_0[x] \ \mathbb{N}\ F'_1[x]$, there exists v'_1 such that $F'_1[x] \Downarrow_{\mathrm{v}} v'_1$ and $x \ \mathbb{N}^{\mathsf{NF}\eta}\ v'_1$. We have $(\overrightarrow{F_0^{(3)}}F''_0, t_0)\ [\mathbb{N}]\ (\overrightarrow{F_1^{(3)}}F''_1, F'_1[t_1])$, so by the induction hypothesis, for a fresh y, there exists t'_0, t'_1 such that $\overrightarrow{F_0^{(3)}}[F''_0[y]] \rightarrow^*_{\mathrm{v}} t'_0$, $\overrightarrow{F_1^{(3)}}[F''_1[y]] \rightarrow^*_{\mathrm{v}} t'_1$, and $t'_0 \ \mathcal{R}\ t'_1$. Hence, we have $\overrightarrow{F_0}[F'_0[v_0]] = \overrightarrow{F_0^{(3)}}[F''_0[v_0]] \rightarrow^*_{\mathrm{v}} t'_0\{x/y\}\{v_0/x\}$, and $\overrightarrow{F_1}[F'_1[v_1]] \rightarrow^*_{\mathrm{v}} \overrightarrow{F_1^{(3)}}[F''_1[v_1]] \rightarrow^*_{\mathrm{v}} t'_1\{v'_1/y\}\{v_1/x\}$, with the resulting terms related by $\mathcal{R}$, as wished.

Lemma A.7. *If $\overrightarrow{F_0}\ [\mathbb{N}]\ \overrightarrow{F_1}$ and $\overrightarrow{\sigma_0}\ \langle\mathbb{N}\rangle\ \overrightarrow{\sigma_1}$ then $\overrightarrow{F_0}[\,]\overrightarrow{\sigma_0}\ \mathcal{R}^{\mathsf{C}\searrow}\ \overrightarrow{F_1}[\,]\overrightarrow{\sigma_1}$.*

Proof. If $\overrightarrow{F_0}[\,]$ is pure, then $\overrightarrow{F_1}[\,]$ is pure by Lemma A.3, and we have $(\overrightarrow{F_0}, x)\ [\mathbb{N}]\ (\overrightarrow{F_1}, x)$ by definition, hence the result holds.

Otherwise, we have $\overrightarrow{F_0} = \overrightarrow{F'_0}F''_0[\langle E'_0\rangle]\overrightarrow{E_0}$, $\overrightarrow{F_1} = \overrightarrow{F'_1}F''_1[\langle E'_1\rangle]\overrightarrow{E_1}$, with $F''_0[x]\ \mathbb{N}\ F''_1[x]$ and $\langle E'_0[x]\rangle\ \mathbb{N}\ \langle E'_1[x]\rangle$ for a fresh x. We have to prove $\langle E'_0[\overrightarrow{E_0}[x]]\rangle\overrightarrow{\sigma_0}\ \mathcal{R}\ \langle E'_1[\overrightarrow{E_1}[x]]\rangle\overrightarrow{\sigma_1}$, and $\overrightarrow{F'_0}[F''_0[x]]\overrightarrow{\sigma_0} \rightarrow^*_{\mathrm{v}} t_0$, $\overrightarrow{F'_1}[F''_1[x]]\overrightarrow{\sigma_1} \rightarrow^*_{\mathrm{v}} t_1$ for some t_0, t_1 such that $t_0\ \mathcal{R}\ t_1$. The former is easy to check, because we necessarily have $(\overrightarrow{E_0}, x)\ [\mathbb{N}]\ (\overrightarrow{E_1}, x)$. The latter is a consequence of Lemma A.6, by taking $v_0 = v_1 = x$.

Lemma A.8. *If $y_0, \dots, y_m$ is a sequence of variables such that $\overrightarrow{\sigma}(y_i) = v_{i+1}$ for $0 \le i \le m-1$ with $y_i\ \mathbb{N}^{\mathsf{NF}\eta}\ v_i$ for $1 \le i \le m$, then for any sequence of fresh variables $z_1, \dots, z_m$ and a value v, there exist a sequence of contexts $E_1, \dots, E_m$ and a sequence of values $v'_0, \dots, v'_m$ such that*

$$(y_0\ v)\overrightarrow{\sigma} \rightarrow^*_{\mathrm{v}} E_1 \dots E_m[y_m\ v'_m]\{v'_{m-1}/z_m\} \dots \{v'_0/z_1\}\overrightarrow{\sigma}$$

where for $1 \le i \le k$, $\Box\ \mathbb{N}^{\mathsf{C}}\ E_i$ and $z_i\ \mathbb{N}^{\mathsf{NF}\eta}\ v'_i$, and also $v'_0 = v$.

Proof. By induction on m.

Lemma A.9. *Let $\overrightarrow{F_0}\ [\mathbb{N}]\ \overrightarrow{F_1}$, $v_0\ \mathbb{N}^{\mathsf{NF}\eta}\ v_1$, $\overrightarrow{\sigma_0}\ \langle\mathbb{N}\rangle\ \overrightarrow{\sigma_1}$, and $v = x\overrightarrow{\sigma_0}$.*

- *If v is a variable, then there exists t_1 such that $\overrightarrow{F_1}[x\ v_1]\overrightarrow{\sigma_1} \Downarrow_{\mathrm{v}} t_1$ and $\overrightarrow{F_0}[x\ v_0]\overrightarrow{\sigma_0}\ \mathcal{R}^{\mathsf{NF}\eta\searrow}\ t_1$.*
- *If v is a λ-abstraction, then there exist t_0, t_1 such that $\overrightarrow{F_0}[x\ v_0]\overrightarrow{\sigma_0} \rightarrow_{\mathrm{v}} t_0$, $\overrightarrow{F_1}[x\ v_1]\overrightarrow{\sigma_1} \rightarrow^*_{\mathrm{v}} t_1$, and $t_0\ \mathcal{R}\ t_1$.*

Proof. There exists a sequence of variables $y_0 = x, \dots, y_m$ such that $\overrightarrow{\sigma_0}(y_i) = y_{i+1}$ for $0 \le i \le m-1$ and $\overrightarrow{\sigma_0}(y_m) = v$. But then $\overrightarrow{\sigma_1}(y_i) = v^1_i$ for $0 \le i \le m$ with $y_i\ \mathbb{N}^{\mathsf{NF}\eta}\ v^1_{i-1}$ for $1 \le i \le m$, and $v\ \langle\mathbb{N}\rangle\ v^1_m$. By Lemma A.8, there exist a sequence of pure contexts $E_1 \dots E_m$, such that $\Box\ \mathbb{N}^{\mathsf{C}}\ E_i$ for $1 \le i \le m$, and a sequence of values $w^1_0, \dots, w^1_m$, such that $z_i\ \mathbb{N}^{\mathsf{NF}\eta}\ w^1_i$ for $1 \le i \le m$ and $w^1_0 = v_1$, with

$$\overrightarrow{F_1}[x\ v_1]\overrightarrow{\sigma_1} \rightarrow^*_{\mathrm{v}} \overrightarrow{F_1}\overrightarrow{E_1}[v^1_m \star z_{m+1}]\{w^1_m/z_{m+1}\}\{w^1_{m-1}/z_m\} \dots \{w^1_0/z_1\}\overrightarrow{\sigma_1},$$

where $z_1, \ldots, z_{m+1}$ are fresh variables. We now distinguish cases based on v.

If $v = y$, then because $y \ \mathbb{N}^{\mathsf{NF}\eta} \ v_m^1$, there exists E_{m+1}^1, w_{m+1}^1 such that $v_m^1 \star z_{m+1} \Downarrow_{\mathrm{v}} E_{m+1}^1[y w_{m+1}^1]$, $\square \ \mathbb{N}^{\mathsf{C}} \ E_{m+1}^1$, and $z_{m+1} \ \mathbb{N}^{\mathsf{NF}\eta} \ w_{m+1}^1$. Consequently, we have

$$\overrightarrow{F_1}[x\, v_1]\overrightarrow{\sigma_1} \rightarrow_{\mathrm{v}}^* \overrightarrow{F_1}\overrightarrow{E_1}[y\, w_{m+1}^1]\{w_m^1/z_{m+1}\}\{w_{m-1}^1/z_m\}\ldots\{w_0^1/z_1\}\overrightarrow{\sigma_1},$$

but we also have

$$\overrightarrow{F_0}[x\, v_0]\overrightarrow{\sigma_0} = \overrightarrow{F_0}\overrightarrow{\square}[y\, z_{m+1}]\{z_m/z_{m+1}\}\ldots\{v_0/z_1\}\overrightarrow{\sigma_0}$$

We can then conclude by Lemmas A.5, and A.7.

If $v = \lambda y.t$, then $\overrightarrow{F_0}[t_0]\overrightarrow{\sigma_0} \rightarrow_{\mathrm{v}} \overrightarrow{F_0}[t\{v_0/y\}]\overrightarrow{\sigma_0}$. The obtained term can be represented as

$$\overrightarrow{F_0}\overrightarrow{\square}[\lambda y.t \star z_{m+1}]\{z_m/z_{m+1}\}\{z_{m-1}/z_m\}\ldots\{v_0/z_1\}\overrightarrow{\sigma_0}.$$

If $\lambda y.t \ \mathbb{N}^{\mathsf{NF}\eta} \ v_m^1$, then the obtained terms are related by $\mathcal{R}$. If $t = \langle E_0''[\overrightarrow{E_0'}[y]]\rangle$ and $v_m^1 = \lambda y.\langle E_1''[\overrightarrow{E_1'}[y]]\rangle$ where $\overrightarrow{E_0'} \ \mathbb{N}^{\mathsf{C}} \ \overrightarrow{E_1'}$ and $\langle E_0''\rangle \ \mathbb{N}^{\mathsf{C}} \ \langle E_1''\rangle$, we also obtained two terms related by $\mathcal{R}$.

Lemma A.10. *If $\overrightarrow{E_0} \ [\mathbb{N}] \ \overrightarrow{E_1}$, $\langle t_0\rangle \ \mathbb{N} \ \langle t_1\rangle$, and $\overrightarrow{\sigma_0} \ \langle\mathbb{N}\rangle \ \overrightarrow{\sigma_1}$ then $\overrightarrow{E_0}[\mathcal{S}k.t_0]\overrightarrow{\sigma_0} \ \mathcal{R}^{\mathsf{NF}\eta\searrow} \ \overrightarrow{E_1}[\mathcal{S}k.t_1]\overrightarrow{\sigma_1}$.*

Proof. From the definition of $\mathcal{R}$, by Lemma A.7.

Lemma A.11. *If $\overrightarrow{E} = E_1 \ldots E_m$ is a sequence of pure contexts such that for each $1 \leq i \leq m$, $E_i \ \mathbb{N}^{\mathsf{C}} \ \square$, then for any fresh variables $z_1, \ldots, z_m$ and value v there exist $v_0 \ldots v_m$ such that $\overrightarrow{E}[v] \Downarrow_{\mathrm{v}} v_0\{v_1/z_1\}\ldots\{v_m/z_m\}$ with $z_i \ \mathbb{N}^{\mathsf{NF}\eta} \ v_{i-1}$ for $1 \leq i \leq m$ and $v_m = v$.*

Proof. By induction on m.

Lemma A.12. *If $\overrightarrow{F_0} \ [\mathbb{N}] \ \overrightarrow{F_1}$, $\overrightarrow{\square} \ [\mathbb{N}] \ \overrightarrow{E_1}$, $v_0 \ \mathbb{N}^{\mathsf{NF}\eta} \ v_1$ and $\overrightarrow{\sigma_0} \ \langle\mathbb{N}\rangle \ \overrightarrow{\sigma_1}$, then for a fresh z, there exist $\overrightarrow{\sigma_0'}$, $\overrightarrow{\sigma_1'}$ such that $\overrightarrow{F_1}[\overrightarrow{E_1}[v_1]]\overrightarrow{\sigma_1} \rightarrow_{\mathrm{v}}^* \overrightarrow{F_1}[z]\overrightarrow{\sigma_1'}$, $\overrightarrow{\sigma_0'} \ \langle\mathbb{N}\rangle \ \overrightarrow{\sigma_1'}$, and $\overrightarrow{F_0}[v_0]\overrightarrow{\sigma_0} = \overrightarrow{F_0}[z]\overrightarrow{\sigma_0'} \ \mathcal{R} \ \overrightarrow{F_1}[z]\overrightarrow{\sigma_1'}$.*

Proof. By Lemma A.11,

$$\overrightarrow{E_1}[v_1]\overrightarrow{\sigma_1} \Downarrow_{\mathrm{v}} v_0^1\{v_1^1/z_1\}\ldots\{v_{m-1}^1/z_{m-1}\}\{v_1/z_m\}\overrightarrow{\sigma_1}$$

where $z_i \ \mathbb{N}^{\mathsf{NF}\eta} \ v_{i-1}^1$ for $1 \leq i \leq m$. The resulting term can be represented as

$$z\{v_0^1/z\}\{v_1^1/z_1\}\ldots\{v_{m-1}^1/z_{m-1}\}\{v_1/z_m\}\overrightarrow{\sigma_1}.$$

We also have $v_0\overrightarrow{\sigma_0} = z\{z_1/z\}\{z_2/z_1\}\ldots\{z_m/z_{m-1}\}\{v_0/z_m\}\overrightarrow{\sigma_0}$ which allows us to conclude.

Lemma A.13. *$\mathcal{R}$ is a normal-form bisimulation up to reduction.*

Proof. By case analysis on $\overrightarrow{F_0}[t_0]\overrightarrow{\sigma_0}$.

Case: $\overrightarrow{F_0}[t_0]\overrightarrow{\sigma_0}$ is a normal form. There are several subcases:

Subcase: $t_0 = F_0[x\, v_0]$ and $x\overrightarrow{\sigma_0} = y$. There exist F_1 and v_1 such that $t_1 \Downarrow_{\mathrm{v}} F_1[x\, v_1]$ with $F_0 \ \mathbb{N}^{\mathsf{C}}\ F_1$ and $v_0 \ \mathbb{N}^{\mathsf{NF}\eta}\ v_1$. Therefore, $\overrightarrow{F_1}[t_1]\overrightarrow{\sigma_1} \to_{\mathrm{v}}^* \overrightarrow{F_1}F_1[x\, v_1]\overrightarrow{\sigma_1}$. We conclude by Lemma A.9.

Subcase: $t_0 = E_0[\mathcal{S}k.t_0']$. Then $\overrightarrow{F_1} = \overrightarrow{E_1}$ and there exist E_1 and t_1' such that $t_1 \Downarrow_{\mathrm{v}} E_1[\mathcal{S}k.t_1']$ with $E_0 \ \mathbb{N}^{\mathsf{C}}\ E_1$ and $\langle t_0'\rangle \ \mathbb{N}\ \langle t_1'\rangle$. Therefore, $\overrightarrow{E_1}[t_1]\overrightarrow{\sigma_1} \to_{\mathrm{v}}^* \overrightarrow{E_1}E_1[\mathcal{S}k.t_1']\overrightarrow{\sigma_1}$. We conclude by Lemma A.10.

Subcase: $t_0 = v_0$. Then there exists v_1 so that $t_1 \Downarrow_{\mathrm{v}} v_1$ and $v_0 \ \mathbb{N}^{\mathsf{NF}\eta}\ v_1$. We also have $\overrightarrow{F_1}[t_1]\overrightarrow{\sigma_1} \to_{\mathrm{v}}^* \overrightarrow{F_1}[v_1]\overrightarrow{\sigma_1}$. If $\overrightarrow{F_0} = \overrightarrow{\Box}$, then we can conclude directly with Lemma A.12. Otherwise, we have $\overrightarrow{F_0} = \overrightarrow{F_0'}F_0''\overrightarrow{\Box}$, with F_0'' not $\Box$; then $\overrightarrow{F_0} = \overrightarrow{F_1'}F_1''\overrightarrow{E_1}$, with $\overrightarrow{F_0'}\ [\mathbb{N}]\ \overrightarrow{F_1'}$, $F_0'' \ \mathbb{N}^{\mathsf{C}}\ F_1''$, and $\overrightarrow{\Box}\ [\mathbb{N}]\ \overrightarrow{E_1}$. By Lemma A.12, for a fresh z, there exist $\overrightarrow{\sigma_0'}$, $\overrightarrow{\sigma_1'}$ such that $\overrightarrow{F_1'}F_1''[\overrightarrow{E_1}[v_1]]\overrightarrow{\sigma_1} \to_{\mathrm{v}}^* \overrightarrow{F_1'}F_1''[z]\overrightarrow{\sigma_1'}$, $\overrightarrow{\sigma_0'}\ \langle\mathbb{N}\rangle\ \overrightarrow{\sigma_1'}$, and $\overrightarrow{F_0'}F_0''[v_0]\overrightarrow{\sigma_0} = \overrightarrow{F_0'}F_0''[z]\overrightarrow{\sigma_0'}\ \mathcal{R}\ \overrightarrow{F_1'}F_1''[z]\overrightarrow{\sigma_1'}$. If $F_0''[z]$ is an open stuck term, we conclude by Lemma A.9; otherwise, $F_0''[z]$ is a control stuck term (and the contexts in $\overrightarrow{F_0'}$, $\overrightarrow{F_1'}$ are pure), and we conclude with Lemma A.10.

Case: $\overrightarrow{F_0}[t_0]\overrightarrow{\sigma_0}$ makes a transition. We have $\overrightarrow{F_0}[t_0]\overrightarrow{\sigma_0} \to_{\mathrm{v}} t_0'$. For each of the following subcases, we prove that there exists t_1' such that $\overrightarrow{F_1}[t_1]\overrightarrow{\sigma_1} \to_{\mathrm{v}}^* t_1'$ and $t_0' \ \mathcal{R}\ t_1'$.

Subcase: $t_0 \to_{\mathrm{v}} t_0'$. We have two possibilities. First, suppose $\overrightarrow{F_0} = \overrightarrow{F_0'}F_0''\overrightarrow{\Box}$, $\overrightarrow{F_1} = \overrightarrow{F_1'}F_1''\overrightarrow{\Box}$, $F_0''[x]\ \mathbb{N}\ F_1''[x]$ for a fresh x, t_0, t_1 delimited, and $t_0 \to_{\mathrm{v}} v_0$. Because $t_0 \ \mathbb{N}\ t_1$, there exists v_1 such that $t_1 \Downarrow_{\mathrm{v}} v_1$ and $v_0 \ \mathbb{N}^{\mathsf{NF}\eta}\ v_1$. We have $\overrightarrow{F_1}[t_1] \to_{\mathrm{v}}^* \overrightarrow{F_1}[v_1]$, and we can conclude using Lemma A.6.

Otherwise, we have $t_1 \ \mathbb{N}\ t_0'$ (because $\to_{\mathrm{v}} \subseteq \mathbb{N}$), therefore $\overrightarrow{F_0}[t_0']\overrightarrow{\sigma_0}\ \mathcal{R}\ \overrightarrow{F_1}[t_1]\overrightarrow{\sigma_1}$ holds directly.

Subcase: $t_0 = E_0[\mathcal{S}k.t_0']$ and $\overrightarrow{F_0} = F[\langle E\rangle]$. Then $\overrightarrow{F_0} = \overrightarrow{F_0'}F_0'[\langle E_0'\rangle]\overrightarrow{E_0'}$ and

$$\begin{aligned}\overrightarrow{F_0}[t_0]\overrightarrow{\sigma_0} &\to_{\mathrm{v}} \overrightarrow{F_0'}F_0'[\langle t_0'\{\lambda x.\langle E_0'\overrightarrow{E_0'}E_0[x]\rangle/k\}\rangle]\overrightarrow{\sigma_0} \\ &= \overrightarrow{F_0'}F_0'[\langle t_0'\rangle]\{\lambda x.\langle E_0'\overrightarrow{E_0'}E_0[x]\rangle/k\}\overrightarrow{\sigma_0}.\end{aligned}$$

By bisimilarity, we know that $t_1 \Downarrow_{\mathrm{v}} E_1[\mathcal{S}k.t_1']$ with $E_0 \ \mathbb{N}^{\mathsf{C}}\ E_1$ and $\langle t_0\rangle \ \mathbb{N}\ \langle t_1\rangle$. We also know that $\overrightarrow{F_1} = \overrightarrow{F_1'}F_1'[\langle E_1'\rangle]\overrightarrow{E_1'}$ where $F_0'[y]\ \mathbb{N}\ F_1'[y]$ for a fresh y, $\langle E_0'\rangle \ \mathbb{N}^{\mathsf{C}}\ \langle E_1'\rangle$, and $\overrightarrow{E_0'}\ [\mathbb{N}]\ \overrightarrow{E_1'}$. Since

$$\begin{aligned}\overrightarrow{F}_1[t_1]\overrightarrow{\sigma_1} &\to_{\mathrm{v}}^* \overrightarrow{F_1'}F_1'[\langle t_1'\{\lambda x.\langle E_1'\overrightarrow{E_1'}E_1[x]\rangle/k\}\rangle]\overrightarrow{\sigma_1} \\ &= \overrightarrow{F_1'}F_1'[\langle t_1'\rangle]\{\lambda x.\langle E_1'\overrightarrow{E_1'}E_1[x]\rangle/k\}\overrightarrow{\sigma_1},\end{aligned}$$

we obtain terms in $\mathcal{R}$, as wished.

Subcase: $t_0 = F'_0[x\ v_0]$ *and* $x\overrightarrow{\sigma_0} = \lambda y.t'_0$. By bisimilarity, $t_1 \Downarrow_{\mathrm{v}} F'_1[x\ v_1]$ with $F'_0 \ \mathbb{N}^{\mathsf{C}}\ F'_1$ and $v_0 \ \mathbb{N}^{\mathsf{NF}\eta}\ v_1$. By Lemma A.9, we know that we obtain terms in $\mathcal{R}$, as wished.

Subcase: $t_0 = v_0$. By bisimilarity, there exists v_1 such that $t_1 \Downarrow_{\mathrm{v}} v_1$ with $v_0 \ \mathbb{N}^{\mathsf{NF}\eta}\ v_1$, and we also have $\overrightarrow{F_1}[t_1]\overrightarrow{\sigma_1} \to^*_{\mathrm{v}} \overrightarrow{F_1}[v_1]\overrightarrow{\sigma_1}$. Suppose $\overrightarrow{F_0} = \overrightarrow{F'_0}F''_0\overrightarrow{\Box}$; then $\overrightarrow{F_1} = \overrightarrow{F'_1}F''_1\overrightarrow{E_1}$, with $\overrightarrow{\Box}\ [\mathbb{N}]\ \overrightarrow{E_1}$ and necessarily $F''_0 \ \mathbb{N}^{\mathsf{C}}\ F''_1$. By Lemma A.11, we have

$$\overrightarrow{F_1}[v_1]\overrightarrow{\sigma_1} \to^*_{\mathrm{v}} \overrightarrow{F'_1}F''_1[z_0]\{v^1_0/z_0\}\{v^1_1/z_1\}\dots\{v^1_k/z_k\}\overrightarrow{\sigma_1}$$

where $z_0, \dots, z_k$ are fresh variables, $z_i \ \mathbb{N}^{\mathsf{NF}\eta}\ v^1_{i-1}$ for $1 \le i \le k$, and $v^1_k = v_1$. But the term $\overrightarrow{F}_0[v_0]\overrightarrow{\sigma_0}$ can also be represented as

$$\overrightarrow{F'_0}F''_0[z_0]\{z_1/z_0\}\{z_2/z_1\}\dots\{v_0/z_k\}\overrightarrow{\sigma_0}.$$

Suppose $F''_0 = F^{(3)}_0[\langle E'_0\rangle]$; then $F''_1 = F^{(3)}_1[\langle E'_1\rangle]$, with $F^{(3)}_0[z_0]\ \mathbb{N}\ F^{(3)}_1[z_0]$ and $\langle E'_0[z_0]\rangle\ \mathbb{N}\ \langle E'_1[z_0]\rangle$. We distinguish cases based on $\langle E'_0[z_0]\rangle$.

- If $\langle E'_0[z_0]\rangle \to_{\mathrm{v}} t'_0$ with t'_0 delimited, then we have $t'_0\ \mathbb{N}\ \langle E'_1[z_0]\rangle$, because $\to_{\mathrm{v}} \subseteq \mathbb{N}$. Therefore, we have

  $$\overrightarrow{F'_0}F^{(3)}_0[t'_0]\{z_1/z_0\}\{z_2/z_1\}\dots\{v_0/z_k\}\overrightarrow{\sigma_0}\ \mathcal{R}$$
  $$\overrightarrow{F'_1}F^{(3)}_1[\langle E'_1[z_0]\rangle]\{v^1_0/z_0\}\{v^1_1/z_1\}\dots\{v^1_k/z_k\}\overrightarrow{\sigma_1}$$

  as wished.

- If $\langle E'_0[z_0]\rangle \to_{\mathrm{v}} v'_0$, then there exists v'_1 such that $\langle E'_1[z_0]\rangle \Downarrow_{\mathrm{v}} v'_1$ and $v'_0 \ \mathbb{N}^{\mathsf{NF}\eta}\ v'_1$. We conclude by using Lemma A.6.

- If $\langle E'_0[z_0]\rangle$ is an open stuck term t'_0, then there exists t'_1 such that $\langle E'_1[z_0]\rangle \Downarrow_{\mathrm{v}} t'_1$ and $t'_0 \ \mathbb{N}^{\mathsf{NF}\eta}\ t'_1$. We conclude by using Lemma A.9.

Suppose F''_0 is a pure context E'_0; then $F''_1 = E'_1$, with $E'_0[z_0]\ \mathbb{N}\ E'_1[z_0]$. We distinguish cases based on $E'_0[z_0]$. If $E'_0[z_0] \to_{\mathrm{v}} t'_0$ or if $E'_0[z_0]$ is an open stuck term, then we conclude as in the previous case. If $E'_0[z_0]$ is a control stuck term $E''_0[\mathcal{S}k.t'_0]$, then there exist E''_1, t''_1 such that $E'_1[z_0] \Downarrow_{\mathrm{v}} E''_1[\mathcal{S}k.t'_0]$, $E''_0 \ \mathbb{N}^{\mathsf{C}}\ E''_1$, and $\langle t'_0\rangle\ \mathbb{N}\ \langle t'_1\rangle$. We conclude as in the case where t_0 is a control stuck term.

□

This gives compatibility of $\mathbb{N}$ w.r.t. evaluation contexts. We can then prove easily compatibility w.r.t. λ-abstraction and shift.

A.2. Bisimulation up to context

$$\frac{}{t \mathrel{\widehat{\mathcal{R}}} t} \qquad \frac{t_0 \mathrel{\mathcal{R}} t_1}{t_0 \mathrel{\widehat{\mathcal{R}}} t_1} \qquad \frac{t_0 \mathrel{\widehat{\mathcal{R}}} t_1 \qquad v_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1}{t_0\{v_0/x\} \mathrel{\widehat{\mathcal{R}}} t_1\{v_1/x\}} \qquad \frac{t_0 \mathrel{\widehat{\mathcal{R}}} t_1 \qquad F_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{C}\searrow}} F_1}{F_0[t_0] \mathrel{\widehat{\mathcal{R}}} F_1[t_1]}$$

$$\frac{t_0 \mathrel{\widehat{\mathcal{R}}} t_1 \qquad t_0, t_1 \text{ delimited} \qquad F_0[x] \mathrel{\widehat{\mathcal{R}}^{\searrow}} F_1[x] \qquad x \text{ fresh}}{F_0[t_0] \mathrel{\widehat{\mathcal{R}}} F_1[t_1]} \qquad \frac{t_0 \mathrel{\widehat{\mathcal{R}}} t_1}{\lambda x.t_0 \mathrel{\widehat{\mathcal{R}}} \lambda x.t_1}$$

$$\frac{t_0 \mathrel{\widehat{\mathcal{R}}} t_1}{\mathcal{S}k.t_0 \mathrel{\widehat{\mathcal{R}}} \mathcal{S}k.t_1}$$

Lemma A.14. *If $F_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{C}\searrow}} F_1$ and $v_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1$, then $F_0\{v_0/x\} \mathrel{\widehat{\mathcal{R}}^{\mathsf{C}\searrow}} F_1\{v_1/x\}$.*

Proof. By definition of $\mathcal{R}^{\mathsf{NF}\searrow}$.

Lemma A.15. *If $t_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} t_1$, $v_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1$, and $t_0\{v_0/x\}$ and $t_1\{v_1/x\}$ are normal forms, then $t_0\{v_0/x\} \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} t_1\{v_1/x\}$.*

Proof. By case analysis on $t_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} t_1$, and using Lemma A.14 in the (open and control) stuck term cases.

Lemma A.16. *If $F_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{C}\searrow}} F_1$ and $F_0' \mathrel{\widehat{\mathcal{R}}^{\mathsf{C}\searrow}} F_1'$, then $F_0[F_0'] \mathrel{\widehat{\mathcal{R}}^{\mathsf{C}\searrow}} F_1[F_1']$.*

Proof. By definition of $\mathcal{R}^{\mathsf{C}\searrow}$.

Lemma A.17. *If $F_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{C}\searrow}} F_1$, $t_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} t_1$, and $F_0[t_0]$ and $F_1[t_1]$ are normal form, then $F_0[t_0] \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} F_1[t_1]$.*

Proof. By case analysis on $F_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{C}\searrow}} F_1$ and $t_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} t_1$, using Lemma A.16 in the (open and control) stuck term cases.

Lemma A.18. *If $\mathcal{R}$ is a bisimulation up to context and $v_0 \mathrel{\widehat{\mathcal{R}}} t_1$, then there exists v_1 such that $t_1 \Downarrow_{\mathrm{v}} v_1$ and $v_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1$.*

Proof. By induction on the derivation of $v_0 \mathrel{\widehat{\mathcal{R}}} t_1$. If $v_0 = t_1$, then the result is direct. If $v_0 \mathrel{\mathcal{R}} t_1$, then the result is a consequence of $\mathcal{R}$ being a bisimulation up to context. The result is also direct in the case of compatibility w.r.t. λ-abstraction. The case $v_0 = v_0'\{v_0''/x\}$, $t_1 = t_1'\{v_1''/x\}$ with $v_0' \mathrel{\widehat{\mathcal{R}}} t_1'$ and $v_0'' \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1''$ is straightforward by induction.

If $t_1 = E_1[t_1']$ with $v_0 \mathrel{\widehat{\mathcal{R}}} t_1'$ and $x \mathrel{\widehat{\mathcal{R}}} E_1[x]$, then by the induction hypothesis, there exists v_1 such that $t_1' \Downarrow_{\mathrm{v}} v_1$ with $v_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1$, and we also have $E_1[x] \Downarrow_{\mathrm{v}} x$. As a result, we have $t_1 \Downarrow_{\mathrm{v}} v_1$ with $v_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1$, as wished.

Lemma A.19. *If $\mathcal{R}$ is a bisimulation up to context, $F_0[x] \widehat{\mathcal{R}}^{\searrow} F_1[x]$ for a fresh x, $t_0 \widehat{\mathcal{R}}^{\mathsf{NF}\searrow} t_1$, and t_0, t_1 are delimited open stuck terms, then $F_0[t_0] \widehat{\mathcal{R}}^{\mathsf{NF}\searrow} F_1[t_1]$.*

Proof. Because t_0 and t_1 are delimited open stuck terms, we have $t_0 = F'_0[\langle E_0[y\, v_0]\rangle]$, $t_1 = F'_1[\langle E_1[y\, v_1]\rangle]$, with $F'_0[z] \widehat{\mathcal{R}}^{\searrow} F'_1[z]$, $\langle E_0[z]\rangle \widehat{\mathcal{R}} \langle E_1[z]\rangle$ for a fresh z, and $v_0 \widehat{\mathcal{R}}^{\mathsf{NF}\searrow} v_1$. To conclude, we have to prove that $F_0[F'_0[z]] \widehat{\mathcal{R}}^{\searrow} F_1[F'_1[z]]$ holds. We know there exist t'_0, t'_1, such that $F'_0[y] \rightarrow^*_{\mathrm{v}} t'_0$, $F'_1[y] \rightarrow^*_{\mathrm{v}} t'_1$, and $t'_0 \widehat{\mathcal{R}} t'_1$. Note that F'_0 and F'_1 are either delimited or $\square$, therefore t'_0 and t'_1 are either delimited or values. If t'_0 and t'_1 are both delimited or both values, then we have $F_0[t'_0] \widehat{\mathcal{R}} F_1[t'_1]$, as wished. Otherwise, we can assume without loss of generality that t'_0 is a value v'_0 and t'_1 is delimited. By Lemma A.18, there exists v'_1 such that $t'_1 \Downarrow_{\mathrm{v}} v'_1$ and $v'_0 \widehat{\mathcal{R}}^{\mathsf{NF}\searrow} v'_1$. As a result, we have $F_1[F'_1[z]] \rightarrow^*_{\mathrm{v}} F_1[v'_1]$ with $F_0[x]\{v_0/x\} \widehat{\mathcal{R}} F_1[x]\{v_1/x\}$, hence the result holds.

Lemma A.20. *Let $\mathcal{R}$ be a bisimulation up to context, and $t_0 \widehat{\mathcal{R}} t_1$ such that t_0 is a normal form. There exists t'_1 such that $t_1 \Downarrow_{\mathrm{v}} t'_1$ and $t_0 \widehat{\mathcal{R}}^{\mathsf{NF}\searrow} t'_1$.*

Proof. By induction on the derivation of $t_0 \widehat{\mathcal{R}} t_1$. The identity, bisimulation, compatibility w.r.t. λ-abstraction and shift cases are straightforward. The substitutivity case is a consequence of Lemma A.15, and the two cases of compatibility w.r.t. context are consequences of Lemmas A.17 and A.19.

Lemma A.21. *If $\mathcal{R}$ is a bisimulation up to context, then $\widehat{\mathcal{R}}$ is a non-η bisimulation up to reduction.*

Proof. Because $\widehat{\mathcal{R}}$ is symmetric, we only have to prove that it is a non-η simulation up to reduction. We consider $t_0 \widehat{\mathcal{R}} t_1$ with t_0 evaluating to t'_0 in m steps; we prove that t_1 evaluates to a term t'_1 so that $t'_0 \widehat{\mathcal{R}}^{\mathsf{NF}\searrow} t'_1$. We proceed by induction on m, and on the derivation of $t_0 \widehat{\mathcal{R}} t_1$, ordered lexicographically. Normal forms (i.e., $m = 0$) are dealt with Lemma A.20. Suppose $m > 0$.

Case: $t_0 = t_1$, i.e., $t_0 \widehat{\mathcal{R}} t_0$. The result holds directly.

Case: $t_0 \mathcal{R} t_1$. The result follows from the definition of bisimulation up to context.

Case: $t_0\{v_0/x\} \widehat{\mathcal{R}} t_1\{v_1/x\}$ with $t_0 \widehat{\mathcal{R}} t_1$ and $v_0 \widehat{\mathcal{R}}^{\mathsf{NF}\searrow} v_1$. We distinguish cases based on how $t_0\{v_0/x\}$ reduces to a normal form t'_0.

First, assume t_0 itself reduces to a normal form t''_0 so that $t'_0 = t''_0\{v_0/x\}$. Note that if t''_0 is an open stuck term $F_0[y\, v'_0]$, then necessarily $y \neq x$. By the induction hypothesis, there exists a normal form t''_1 such that $t_1 \Downarrow_{\mathrm{v}} t''_1$ and $t''_0 \widehat{\mathcal{R}}^{\mathsf{NF}\searrow} t''_1$. In particular, if t''_1 is an open stuck term $F_1[z\, v'_1]$, then $z = y \neq x$, and therefore $t''_1\{v_1/x\}$ is a normal form. By Proposition 2.10, we have $t_1\{v_1/x\} \Downarrow_{\mathrm{v}} t''_1\{v_1/x\}$, and $t''_0\{v_0/x\} \widehat{\mathcal{R}}^{\mathsf{NF}\searrow} t''_1\{v_1/x\}$ holds by Lemma A.15. Therefore, we have the required result.

The other possibility is $t_0 \Downarrow_{\mathrm{v}} F_0[x\, v_0'']$ with $F_0\{v_0/x\}[v_0\, v_0''\{v_0/x\}] \Downarrow_{\mathrm{v}} t_0'$. By the induction hypothesis, there exist F_1, v_1'' such that $t_1 \Downarrow_{\mathrm{v}} F_1[x\, v_1'']$, $F_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{C}\searrow}} F_1$, and $v_0'' \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1''$. From $F_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{C}\searrow}} F_1$, $v_0'' \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1''$, and $v_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1$, we deduce $F_0\{v_0/x\} \mathrel{\widehat{\mathcal{R}}^{\mathsf{C}\searrow}} F_1\{v_1/x\}$ (*) and $v_0''\{v_0/x\} \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1''\{v_1/x\}$ (**) by Lemmas A.14 and A.15. We now distinguish cases, based on the kinds of the values v_0 and v_1.

- If v_0 and v_1 are variables, then they are equal since we have $v_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1$. Let $y = v_0 = v_1$. Then we have

  $$t_0\{v_0/x\} \Downarrow_{\mathrm{v}} F_0\{y/x\}[y\, v_0''\{y/x\}]$$
  $$t_1\{v_1/x\} \Downarrow_{\mathrm{v}} F_1\{y/x\}[y\, v_1''\{y/x\}]$$

  We obtain two open stuck terms, and we have $F_0\{y/x\} \mathrel{\widehat{\mathcal{R}}^{\mathsf{C}\searrow}} F_1\{y/x\}$ and $v_0''\{y/x\} \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1''\{y/x\}$ according to (*) and (**), hence the required result holds.

- If v_0 and v_1 are λ-abstractions $\lambda y.s_0$ and $\lambda y.s_1$, then by definition of $\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}$, we have $s_0 \mathrel{\widehat{\mathcal{R}}} s_1$. We have the following reductions.

  $$F_0\{v_0/x\}[v_0\, v_0''\{v_0/x\}] \rightarrow_{\mathrm{v}} F_0\{v_0/x\}[s_0\{v_0''\{v_0/x\}/y\}]$$
  $$F_1\{v_1/x\}[v_1\, v_1''\{v_1/x\}] \rightarrow_{\mathrm{v}} F_1\{v_1/x\}[s_1\{v_1''\{v_1/x\}/y\}]$$

  From (**) and $s_0 \mathrel{\widehat{\mathcal{R}}} s_1$, we deduce $s_0\{v_0''\{v_0/x\}/y\} \mathrel{\widehat{\mathcal{R}}} s_1\{v_1''\{v_1/x\}/y\}$ by definition of $\widehat{\mathcal{R}}$. This result and (*) implies $F_0\{v_0/x\}[s_0\{v_0''\{v_0/x\}/y\}] \mathrel{\widehat{\mathcal{R}}} F_1\{v_1/x\}[s_1\{v_1''\{v_1/x\}/y\}]$ by definition of $\widehat{\mathcal{R}}$. We know that the term $F_0\{v_0/x\}[s_0\{v_0''\{v_0/x\}/y\}]$ evaluates to t_0' in less than $m-1$ steps, therefore we can apply the induction hypothesis: there exists t_1' such that $F_1\{v_1/x\}[s_1\{v_1''\{v_1/x\}/y\}] \Downarrow_{\mathrm{v}} t_1'$ and $t_0' \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} t_1'$. One can check that we have $t_1\{v_1/x\} \Downarrow_{\mathrm{v}} t_1'$, hence the result holds.

Case: $E_0[t_0] \mathrel{\widehat{\mathcal{R}}} E_1[t_1]$ *with* $t_0 \mathrel{\widehat{\mathcal{R}}} t_1$ *and* $E_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{C}\searrow}} E_1$. We distinguish cases based on how $E_0[t_0]$ reduces to a normal form t_0'.

The first possibility is $t_0 \Downarrow_{\mathrm{v}} v_0$ and $E_0[v_0] \Downarrow_{\mathrm{v}} t_0'$. By the induction hypothesis, there exists v_1 such that $t_1 \Downarrow_{\mathrm{v}} v_1$, and $v_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1$. Because $E_0[v_0] \Downarrow_{\mathrm{v}} t_0'$, there exists a normal form t_0'' such that $E_0[x] \Downarrow_{\mathrm{v}} t_0''$ for a fresh x. From $E_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{C}\searrow}} E_1$, using the induction hypothesis, we deduce that there exists a normal form t_1'' such that $E_1[x] \Downarrow_{\mathrm{v}} t_1''$ and $t_0'' \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} t_1''$. By Proposition 2.10, we have $E_0[t_0] \rightarrow_{\mathrm{v}}^* t_0''\{v_0/x\}$ and $E_1[t_1] \rightarrow_{\mathrm{v}}^* t_1''\{v_1/x\}$. So far, we have $E_0[t_0] \rightarrow_{\mathrm{v}}^* t_0''\{v_0/x\} \Downarrow_{\mathrm{v}} t_0'$, and $E_1[t_1] \rightarrow_{\mathrm{v}}^* t_1''\{v_1/x\}$. From $t_0'' \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} t_1''$, $v_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1$, and $t_0''\{v_0/x\} \Downarrow_{\mathrm{v}} t_0'$, we can conclude exactly as in the previous case about substituvity.

The other possibility is $t_0 \Downarrow_{\mathrm{v}} t_0''$, t_0'' is a normal form but not a value, and $t_0' = E_0[t_0'']$. By the induction hypothesis, there exists t_1'' such that $t_1 \Downarrow_{\mathrm{v}} t_1''$ and

$t_0'' \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} t_1''$; t_1'' is also not a value. We have $E_0[t_0] \Downarrow_{\mathrm{v}} E_0[t_0'']$, $E_1[t_1] \Downarrow_{\mathrm{v}} E_1[t_1'']$, and $E_0[t_0''] \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} E_1[t_1'']$ by Lemma A.17, hence the result holds.

Case: $F_0[\langle E_0[t_0]\rangle] \mathrel{\widehat{\mathcal{R}}} F_1[\langle E_1[t_1]\rangle]$ *with* $F_0[x] \mathrel{\widehat{\mathcal{R}}^{\searrow}} F_1[x]$, $\langle E_0[x]\rangle \mathrel{\widehat{\mathcal{R}}} \langle E_1[x]\rangle$ *(x fresh), and* $t_0 \mathrel{\widehat{\mathcal{R}}} t_1$. We distinguish cases based on how $F_0[\langle E_0[t_0]\rangle]$ reduces to a normal form t_0'.

1. Suppose $t_0 \Downarrow_{\mathrm{v}} t_0''$, so that t_0'' is an open stuck term. Therefore, we have $t_0' = F_0[\langle E_0[t_0'']\rangle]$. By the induction hypothesis, there exists an open stuck term t_1'' such that $t_1 \Downarrow_{\mathrm{v}} t_1''$ and $t_0'' \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} t_1''$. We therefore have $F_1[\langle E_1[t_1]\rangle] \rightarrow_{\mathrm{v}}^* F_1[\langle E_1[t_1'']\rangle]$, $F_1[\langle E_1[t_1'']\rangle]$ is an open stuck term, and $F_0[\langle E_0[t_0'']\rangle] \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} F_1[\langle E_1[t_1'']\rangle]$ holds by Lemma A.17. We therefore have the required result.
2. Suppose $t_0 \Downarrow_{\mathrm{v}} v_0$. By the induction hypothesis, there exists v_1 such that $t_1 \Downarrow_{\mathrm{v}} v_1$ and $v_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1$. We know that $\langle E_0[v_0]\rangle$ must reduce to a normal form t_0'', which is either a value or an open stuck term. We have $\langle E_0[x]\rangle \mathrel{\widehat{\mathcal{R}}} \langle E_1[x]\rangle$ for a fresh x, and $v_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1$, so with the same reasoning as in the substitutivity case, we know there exists t_1'' such that $\langle E_1[v_1]\rangle \Downarrow_{\mathrm{v}} t_1''$ and $t_0'' \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} t_1''$. We now distinguish cases based on the kind of t_0'', t_1''.
   (a) Suppose t_0'' and t_1'' are values v_0' and v_1'. We have $F_0[x] \mathrel{\widehat{\mathcal{R}}^{\searrow}} F_1[x]$ for a fresh x, $v_0' \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1'$, and $F_0[v_0'] \Downarrow_{\mathrm{v}} t_0'$, so using the same reasoning as in the substitutivity case, we know there exists t_1' such that $F_1[v_1'] \Downarrow_{\mathrm{v}} t_1'$ and $t_0' \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} t_1'$. We have $F_1[\langle E_1[t_1]\rangle] \rightarrow_{\mathrm{v}}^* F_1[\langle E_1[v_1]\rangle] \rightarrow_{\mathrm{v}}^* F_1[v_1'] \Downarrow_{\mathrm{v}} t_1'$, hence the result holds.
   (b) Suppose t_0'' and t_1'' are open stuck terms. Then $t_0' = F_0[t_0'']$. We have $F_1[\langle E_1[t_1]\rangle] \Downarrow_{\mathrm{v}} F_1[t_1'']$, and $F_0[t_0''] \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} F_1[t_1'']$ holds by Lemma A.19.
3. Suppose $t_0 \Downarrow_{\mathrm{v}} E_0'[\mathcal{S}k.t_0'']$. By the induction hypothesis, there exists a control stuck term $E_1'[\mathcal{S}k.t_1'']$ such that $t_1 \Downarrow_{\mathrm{v}} E_1'[\mathcal{S}k.t_1'']$ and $E_0'[\mathcal{S}k.t_0''] \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} E_1'[\mathcal{S}k.t_1'']$. We know $\langle E_0[E_0'[\mathcal{S}k.t_0'']]\rangle$ must reduce to a normal form s_0, and we have $\langle E_0[E_0'[\mathcal{S}k.t_0'']]\rangle \rightarrow_{\mathrm{v}} \langle t_0''\{\lambda x.\langle E_0[E_0'[x]]\rangle/k\}\rangle$. Consequently, $\langle t_0''\{\lambda x.\langle E_0[E_0'[x]]\rangle/k\}\rangle$ reduces to s_0 in $m-1$ steps or less. Because $\langle t_0''\{\lambda x.\langle E_0[E_0'[x]]\rangle/k\}\rangle \mathrel{\widehat{\mathcal{R}}} \langle t_1''\{\lambda x.\langle E_1[E_1'[x]]\rangle/k\}\rangle$ holds, using the induction hypothesis on m, we know there exists a normal form s_1 such that $\langle t_1''\{\lambda x.\langle E_1[E_1'[x]]\rangle/k\}\rangle \Downarrow_{\mathrm{v}} s_1$ and $s_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} s_1$. We have $F_0[\langle E_0[t_0]\rangle] \rightarrow_{\mathrm{v}}^* F_0[s_0]$ and $F_1[\langle E_1[t_1]\rangle] \rightarrow_{\mathrm{v}}^* F_1[s_1]$. From there, we conclude as before by case analysis on the kind of s_0 and s_1 (either value or open stuck term).

Case: $F_0[t_0] \mathrel{\widehat{\mathcal{R}}} F_1[t_1]$ *with* $F_0[x] \mathrel{\widehat{\mathcal{R}}^{\searrow}} F_1[x]$ *(x fresh),* $t_0 \mathrel{\widehat{\mathcal{R}}} t_1$*, and* t_0*,* t_1 *delimited.* We distinguish cases based on how t_0 reduces to a normal form.

If $t_0 \Downarrow_{\mathrm{v}} v_0$, then by the induction hypothesis, there exists v_1 such that $t_1 \Downarrow_{\mathrm{v}} v_1$ and $v_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1$. By definition of $\widehat{\mathcal{R}}^{\searrow}$, there exist t_0', t_1' such that $F_0[x] \rightarrow_{\mathrm{v}}^* t_0'$, $F_1[x] \rightarrow_{\mathrm{v}}^* t_1'$, and $t_0' \mathrel{\widehat{\mathcal{R}}} t_1'$. We have $F_0[t_0] \rightarrow_{\mathrm{v}}^* t_0'\{v_0/x\}$, $F_1[t_1] \rightarrow_{\mathrm{v}}^* t_1'\{v_1/x\}$, with $t'_0\{v_0/x\} \mathrel{\widehat{\mathcal{R}}} t'_1\{v_1/x\}$, so from there we can conclude as in the substitutivity case.

If $t_0 \Downarrow_{\mathrm{v}} t'_0$ with t'_0 an open stuck term, then by the induction hypothesis, there exists t'_1 such that $t_1 \Downarrow_{\mathrm{v}} t'_1$ and $t'_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} t'_1$. We have $F_1[t_1] \Downarrow_{\mathrm{v}} F_1[t'_1]$, and $F_0[t'_0] \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} F_1[t'_1]$ holds by Lemma A.19.

A.3. Refined bisimilarity and refined bisimulation up to context

The proof of soundness of refined bisimilarity is an adaptation of the proofs of Section A.1 and Section A.4. We only point the main differences with these two proofs. We let σ range over regular and context substitutions. We define $\langle\mathbb{R}\rangle$ as

$$\frac{v_0 \mathrel{\mathbb{R}^{\mathsf{RNF}\eta}} v_1}{v_0 \mathrel{\langle\mathbb{R}\rangle} v_1} \qquad\qquad \frac{\overrightarrow{E_0} \mathrel{\mathbb{R}^{\mathsf{C}}} \overrightarrow{E_1} \qquad \langle E'_0\rangle \mathrel{\mathbb{R}^{\mathsf{C}}} \langle E'_1\rangle}{E'_0\overrightarrow{E_0} \mathrel{\langle\mathbb{R}\rangle} E'_1\overrightarrow{E_1}}$$

We use the same notations and constraints on sequences of substitutions and contexts as in Section A.1, and we define $\mathcal{R}'$ as

$$\mathcal{R}' \stackrel{\mathsf{def}}{=} \{(\overrightarrow{F_0}[t_0]\overrightarrow{\sigma_0}, \overrightarrow{F_1}[t_1]\overrightarrow{\sigma_1}) \mid (t_0, \overrightarrow{F_0}) \mathrel{[\mathbb{R}]} (t_1, \overrightarrow{F_1}), \overrightarrow{\sigma_0} \mathrel{\langle\mathbb{R}\rangle} \overrightarrow{\sigma_1}\}$$

Lemma A.22. *$\mathcal{R}'$ is a refined bisimulation up to reduction.*

PROOF. By case analysis on $\overrightarrow{F_0}[t_0]\overrightarrow{\sigma_0}$. We only sketch the subcases where the proof differ from the proofs of Lemma A.13 and Lemma A.33.

The first two subcases are in the case where $\overrightarrow{F_0}[t_0]\overrightarrow{\sigma_0}$ is a normal form.

Subcase: $t_0 = E_0[\mathcal{S}k.t'_0]$, and $\overrightarrow{F_0} = \overrightarrow{E_0}$. Then $\overrightarrow{F_1} = \overrightarrow{E_1}$, and there exist E_1 and t'_1 such that $t_1 \Downarrow_{\mathrm{v}} E_1[\mathcal{S}k.t'_1]$ with $\langle t'_0\{\lambda x.\langle k'\ E_0[x]\rangle/k\}\rangle \mathrel{\mathbb{R}} \langle t'_1\{\lambda x.\langle k'\ E_1[x]\rangle/k\}\rangle$ for a fresh k'. Therefore, $\overrightarrow{E_1}[t_1]\overrightarrow{\sigma_1} \rightarrow^*_{\mathrm{v}} \overrightarrow{E_1}E_1[\mathcal{S}k.t'_1]\overrightarrow{\sigma_1}$, and we have

$$\begin{aligned}\langle t'_0\{\lambda x.\langle k''\ \overrightarrow{E_0}E_0[x]\rangle/k\}\rangle\overrightarrow{\sigma_0} &= \langle t'_0\{\lambda x.\langle k'\ E_0[x]\rangle/k\}\rangle\{k''\ \overrightarrow{E_0}\ \square/k'\}\overrightarrow{\sigma_0}\\ &\mathrel{\mathcal{R}'} \langle t'_1\{\lambda x.\langle k'\ E_1[x]\rangle/k\}\rangle\{k''\ \overrightarrow{E_1}\ \square/k'\}\overrightarrow{\sigma_1}\\ &= \langle t'_1\{\lambda x.\langle k''\ \overrightarrow{E_1}E_1[x]\rangle/k\}\rangle\overrightarrow{\sigma_1}\end{aligned}$$

for a fresh k'', hence the result holds.

Subcase: $t_0 = F_0[\langle k\ v_0\rangle]$, and $k\overrightarrow{\sigma_0} = k'\,\square$. Similar to the corresponding subcase in the proof of Lemma A.33.

The next subcases are in the case where $\overrightarrow{F_0}[t_0]\overrightarrow{\sigma_0}$ reduces (in one step) to a term t'_0, and we prove that $\overrightarrow{F_1}[t_1]\overrightarrow{\sigma_1}$ reduces (in 0 or more steps) to a t'_1 such that $t'_0 \mathrel{\mathcal{R}'} t'_1$.

Subcase: $t_0 = E_0[\mathcal{S}k.t'_0]$ *and* $\overrightarrow{F_0} = F[\langle E \rangle]$. Then $\overrightarrow{F_0} = \overrightarrow{F'_0} F'_0[\langle E'_0 \rangle] \overrightarrow{E'_0}$ and

$$\begin{aligned}\overrightarrow{F_0}[t_0]\overrightarrow{\sigma_0} \rightarrow_{\mathrm{v}} & \overrightarrow{F'_0} F'_0[\langle t'_0\{\lambda x.\langle E'_0 \overrightarrow{E'_0} E_0[x]\rangle/k\}\rangle]\overrightarrow{\sigma_0} \\ &= \overrightarrow{F'_0} F'_0[\langle t'_0\{\lambda x.\langle k'\ E_0[x]\rangle/k\}\rangle]\{E'_0\overrightarrow{E'_0}/k'\}\overrightarrow{\sigma_0}\end{aligned}$$

for a fresh k'. By bisimilarity, we know there exists $E_1[\mathcal{S}k.t'_1]$ such that $t_1 \Downarrow_{\mathrm{v}} E_1[\mathcal{S}k.t'_1]$ with $\langle t'_0\{\lambda x.\langle k'\ E_0[x]\rangle/k\}\rangle \ \mathbb{R}\ \langle t'_1\{\lambda x.\langle k'\ E_1[x]\rangle/k\}\rangle$. We also know that $\overrightarrow{F_1} = \overrightarrow{F'_1} F'_1[\langle E'_1 \rangle]\overrightarrow{E'_1}$ where $F'_0[y] \ \mathbb{R}\ F'_1[y]$ for a fresh y, $\langle E'_0 \rangle \ \mathbb{R}^{\mathsf{C}}\ \langle E'_1 \rangle$, and $\overrightarrow{E'_0} \ \mathbb{R}^{\mathsf{C}}\ \overrightarrow{E'_1}$. Since

$$\begin{aligned}\overrightarrow{F}_1[t_1]\overrightarrow{\sigma_1} \rightarrow_{\mathrm{v}}^{*} & \overrightarrow{F'_1} F'_1[\langle t'_1\{\lambda x.\langle E'_1 \overrightarrow{E'_1} E_1[x]\rangle/k\}\rangle]\overrightarrow{\sigma_1} \\ &= \overrightarrow{F'_1} F'_1[\langle t'_1\{\lambda x.\langle k'\ E_1[x]\rangle/k\}\rangle]\{E'_1\overrightarrow{E'_1}/k'\}\overrightarrow{\sigma_1},\end{aligned}$$

we get terms in $\mathcal{R}$, as wished.

Subcase $t_0 = F_0[\langle k\ v_0 \rangle]$ *and* $k\overrightarrow{\sigma_0} = E_0$. Similar to the corresponding subcase in the proof of Lemma A.33.

We now adapt the proof of Section A.2 to refined bisimulation up to context. We remind that $\breve{\mathcal{R}}$ is defined as $\widehat{\mathcal{R}}$ but we add the following rule:

$$\frac{t_0\ \breve{\mathcal{R}}\ t_1 \qquad \langle E_0 \rangle\ \breve{\mathcal{R}}^{\mathsf{C}\searrow}\ \langle E_1 \rangle}{t_0\{E_0/k\}\ \breve{\mathcal{R}}\ t_1\{E_1/k\}}$$

Lemma A.23. *If* $\mathcal{R}$ *is a refined bisimulation up to context, then* $\breve{\mathcal{R}}$ *is a non-eta refined bisimulation up to reduction.*

Proof. We consider $t_0\ \breve{\mathcal{R}}\ t_1$ with t_0 evaluating to t'_0 in m steps; we prove that t_1 evaluates to a term t'_1 so that $t'_0\ \breve{\mathcal{R}}^{\mathsf{RNF}\searrow}\ t'_1$. We proceed by induction on m, and on the derivation of $t_0\ \breve{\mathcal{R}}\ t_1$, ordered lexicographically. We only discuss the cases where the proof differs from the proof of Lemma A.21.

Case: $\mathcal{S}k.t_0\ \breve{\mathcal{R}}\ \mathcal{S}k.t_1$ *with* $t_0\ \breve{\mathcal{R}}\ t_1$. We have to prove that $\langle t_0\{\lambda x.\langle k'\ x\rangle/k\}\rangle\ \breve{\mathcal{R}}\ \langle t_1\{\lambda x.\langle k'\ x\rangle/k\}\rangle$ holds for a fresh k'. Because $\breve{\mathcal{R}}$ is reflexive, we have $\lambda x.\langle k'\ x\rangle\ \breve{\mathcal{R}}\ \lambda x.\langle k'\ x\rangle$; we can then conclude using substitutivity and compatibility w.r.t. $\langle . \rangle$.

Case: $E_0[t_0]\ \breve{\mathcal{R}}\ E_1[t_1]$ *with* $E_0\ \breve{\mathcal{R}}^{\mathsf{C}\searrow}\ E_1$ *and* $t_0\ \breve{\mathcal{R}}\ t_1$. We discuss only the case where $t_0 \Downarrow_{\mathrm{v}} E'_0[\mathcal{S}k.t'_0]$. By the induction hypothesis, there exists $E'_1[\mathcal{S}k.t'_1]$ such that $t_1 \Downarrow_{\mathrm{v}} E'_1[\mathcal{S}k.t'_1]$ and $\langle t'_0\{\lambda x.\langle k'\ E'_0[x]\rangle/k\}\rangle\ \breve{\mathcal{R}}\ \langle t'_1\{\lambda x.\langle k'\ E'_1[x]\rangle/k\}\rangle$ for a fresh k'. This implies

$$\langle t'_0\{\lambda x.\langle k'\ E'_0[x]\rangle/k\}\rangle\{k''\ E_0/k'\}\ \breve{\mathcal{R}}\ \langle t'_1\{\lambda x.\langle k'\ E'_1[x]\rangle/k\}\rangle\{k''\ E_1/k'\}$$

for a fresh k'', i.e., $\langle t'_0\{\lambda x.\langle k''\ E_0[E'_0[x]]\rangle/k\}\rangle\ \breve{\mathcal{R}}\ \langle t'_1\{\lambda x.\langle k''\ E_1[E'_1[x]]\rangle/k\}\rangle$. Because we have $E_0[t_0] \Downarrow_{\mathrm{v}} E_0[E'_0[\mathcal{S}k.t'_0]]$ and $E_1[t_1] \Downarrow_{\mathrm{v}} E_1[E'_1[\mathcal{S}k.t'_1]]$, we have the required result.

Case: $F_0[\langle E_0[t_0]\rangle] \mathrel{\breve{\mathcal{R}}} F_1[\langle E_1[t_1]\rangle]$ *with* $F_0[x] \mathrel{\breve{\mathcal{R}}} F_1[x]$, $\langle E_0[x]\rangle \mathrel{\breve{\mathcal{R}}} \langle E_1[x]\rangle$ *(x fresh), and* $t_0 \mathrel{\breve{\mathcal{R}}} t_1$. We discuss only the case where $t_0 \Downarrow_{\mathrm{v}} E_0'[\mathcal{S}k.t_0']$. By the induction hypothesis, there exists $E_1'[\mathcal{S}k.t_1']$ such that $t_1 \Downarrow_{\mathrm{v}} E_1'[\mathcal{S}k.t_1']$ and $\langle t_0'\{\lambda x.\langle k'\, E_0'[x]\rangle/k\}\rangle \mathrel{\breve{\mathcal{R}}} \langle t_1'\{\lambda x.\langle k'\, E_1'[x]\rangle/k\}\rangle$ for a fresh k'. This implies

$$\langle t_0'\{\lambda x.\langle k'\, E_0'[x]\rangle/k\}\rangle\{E_0/k'\} \mathrel{\breve{\mathcal{R}}} \langle t_1'\{\lambda x.\langle k'\, E_1'[x]\rangle/k\}\rangle\{E_1/k'\},$$

which in turn implies $F_0[\langle t_0'\{\lambda x.\langle E_0[E_0'[x]]\rangle/k\}\rangle] \mathrel{\breve{\mathcal{R}}} F_1[\langle t_1'\{\lambda x.\langle E_1[E_1'[x]]\rangle/k\}\rangle]$. We also have $F_0[\langle E_0[t_0]\rangle] \rightarrow_{\mathrm{v}}^* F_0[\langle t_0'\{\lambda x.\langle E_0[E_0'[x]]\rangle/k\}\rangle]$ and $F_1[\langle E_1[t_1]\rangle] \rightarrow_{\mathrm{v}}^* F_1[\langle t_1'\{\lambda x.\langle E_1[E_1'[x]]\rangle/k\}\rangle]$. From there, we can conclude as in the proof of Lemma A.21.

Case: $t_0\{E_0/k\} \mathrel{\breve{\mathcal{R}}} t_1\{E_1/k\}$ *with* $t_0 \mathrel{\breve{\mathcal{R}}} t_1$ *and* $\langle E_0\rangle \mathrel{\breve{\mathcal{R}}^{\mathsf{C}\searrow}} \langle E_1\rangle$. We distinguish two cases. First, suppose $t_0 \Downarrow_{\mathrm{v}} t_0'$, where t_0' is a normal form distinct from $F_0[\langle k\, v_0\rangle]$. Then by the induction hypothesis, there exists t_1' such that $t_1 \Downarrow_{\mathrm{v}} t_1'$ and $t_0' \mathrel{\breve{\mathcal{R}}^{\mathsf{RNF}\searrow}} t_1'$. We therefore have $t_0\{E_0/k\} \Downarrow_{\mathrm{v}} t_0'\{E_0/k\}$, $t_1\{E_1/k\} \Downarrow_{\mathrm{v}} t_1'\{E_1/k\}$, and $t_0'\{E_0/k\} \mathrel{\breve{\mathcal{R}}^{\mathsf{RNF}\searrow}} t_1'\{E_1/k\}$, as required.

Next, suppose $t_0 \Downarrow_{\mathrm{v}} F_0[\langle k\, v_0\rangle]$. Then by the induction hypothesis, there exists $F_1[\langle k\, v_1\rangle]$ such that $t_1 \Downarrow_{\mathrm{v}} F_1[\langle k\, v_1\rangle]$, $F_0[\langle\Box\rangle] \mathrel{\breve{\mathcal{R}}^{\mathsf{C}\searrow}} F_1[\langle\Box\rangle]$, and $v_0 \mathrel{\breve{\mathcal{R}}^{\mathsf{RNF}\searrow}} v_1$. Therefore, we have $t_0\{E_0/k\} \rightarrow_{\mathrm{v}}^* F_0\{E_0/k\}[\langle E_0[v_0\{E_0/k\}]\rangle]$ as well as $t_1\{E_1/k\} \rightarrow_{\mathrm{v}}^* F_1\{E_1/k\}[\langle E_1[v_1\{E_1/k\}]\rangle]$. Because $F_0\{E_0/k\}[\langle E_0[v_0\{E_0/k\}]\rangle] \mathrel{\breve{\mathcal{R}}} F_1\{E_1/k\}[\langle E_1[v_1\{E_1/k\}]\rangle]$, we can conclude from there by case analysis on how $F_0\{E_0/k\}[\langle E_0[v_0\{E_0/k\}]\rangle]$ reduces to a normal form.

A.4. Delimited normal-form bisimilarity

The proof is an adaptation of the one for the relaxed semantics. We let σ range over regular and context substitutions. We use the same notations on sequences of contexts as in Section A.1, and we define $(\overrightarrow{F_0}, t_0)\ [\mathbb{M}]\ (\overrightarrow{F_1}, t_1)$ by the following rules:

$$\frac{t_0 \mathrel{\mathbb{M}} t_1 \qquad t_0, t_1 \text{ delimited}}{(\emptyset, t_0)\ [\mathbb{M}]\ (\emptyset, t_1)} \qquad \frac{F_0'[x] \mathrel{\mathbb{M}} F_1'[x] \quad x \text{ fresh} \qquad (\overrightarrow{F_0}, t_0)\ [\mathbb{M}]\ (\overrightarrow{F_1}, t_1) \quad F_0'\overrightarrow{F_0}[t_0], F_1'\overrightarrow{F_1}[t_1] \text{ delimited}}{(F_0'\overrightarrow{F_0}, t_0)\ [\mathbb{M}]\ (F_1'\overrightarrow{F_1}, t_1)}$$

We write $\overrightarrow{F_0}\ [\mathbb{M}]\ \overrightarrow{F_1}$ if $(\overrightarrow{F_0}, \langle x\rangle)\ [\mathbb{M}]\ (\overrightarrow{F_1}, \langle x\rangle)$ holds for a fresh x. We implicitly use the following properties when manipulating contexts.

Lemma A.24. *Let* $\overrightarrow{F_0} = F_0^1 \dots F_0^m$, $\overrightarrow{F_1} = F_1^1 \dots F_1^m$.

- $(\overrightarrow{F_0}, t_0)\ [\mathbb{M}]\ (\overrightarrow{F_1}, t_1)$ *holds for some delimited* t_0, t_1 *iff for all* $1 \le i \le m$, *either* $F_0^i = F_1^i = \Box$, *or* F_0^i, F_1^i *are both delimited, or one of them is delimited and the other is* $\Box$.
- $\overrightarrow{F_0}\ [\mathbb{M}]\ \overrightarrow{F_1}$ *iff for all* $1 \le i \le m$, $F_0^i\ [\mathbb{M}]\ F_1^i$.

Proof. By induction on m.

We write $\{E_0/k\}\ \langle\mathbb{M}\rangle\ \{E_1/k\}$ (resp. $\{v_0/x\}\ \langle\mathbb{M}\rangle\ \{v_1/x\}$) if $\langle E_0\rangle\ \mathbb{M}^{\mathsf{C}}\ \langle E_1\rangle$ (resp. $v_0\ \mathbb{M}^{\mathsf{NF}\eta}\ v_1$). We use the same notations and constraints on sequences of substitutions as in Section A.1, and we define $\mathcal{R}''$ as

$$\begin{aligned}\mathcal{R}'' \stackrel{\mathsf{def}}{=} &\{(\overrightarrow{F_0}[t_0]\overrightarrow{\sigma_0}, \overrightarrow{F_1}[t_1]\overrightarrow{\sigma_1}) \mid (\overrightarrow{F_0}, t_0)\ [\mathbb{M}]\ (\overrightarrow{F_1}, t_1), \overrightarrow{\sigma_0}\ \langle\mathbb{M}\rangle\ \overrightarrow{\sigma_1}\}\\ &\cup \{(v_0\overrightarrow{\sigma_0}, v_1\overrightarrow{\sigma_1}) \mid v_0\ \mathbb{M}^{\mathsf{NF}\eta}\ v_1, \overrightarrow{\sigma_0}\ \langle\mathbb{M}\rangle\ \overrightarrow{\sigma_1}\}\end{aligned}$$

Lemma A.25. *If $v_0\ \mathbb{M}\ v_1$, then $v_0\ \mathbb{M}^{\mathsf{NF}\eta}\ v_1$.*

Proof. Straightforward by unfolding the definition of $v_0\ \mathbb{M}\ v_1$.

Lemma A.26. *If $t_0\ \mathbb{M}\ t_1$, then for any fresh k, we have $\langle k\, t_0\rangle\ \mathbb{M}\ \langle k\, t_1\rangle$.*

Proof. The result is obvious if t_0 and t_1 are not both delimited. For delimited terms, we prove that $\mathcal{R} \stackrel{\mathsf{def}}{=} \{(\langle k\, t_0\rangle, \langle k\, t_1\rangle) \mid t_0\ \mathbb{M}\ t_1, t_0, t_1 \text{ delimited}\} \cup \mathbb{M}$ is a delimited bisimulation. We proceed by case analysis on t_0.

If $t_0 \Downarrow_{\mathrm{v}} v_0$, then by bisimilarity, there exists v_1 such that $t_1 \Downarrow_{\mathrm{v}} v_1$ and $v_0\ \mathbb{M}^{\mathsf{NF}\eta}\ v_1$. Consequently, we have $\langle k\, t_0\rangle \Downarrow_{\mathrm{v}} \langle k\, v_0\rangle$, $\langle k\, t_1\rangle \Downarrow_{\mathrm{v}} \langle k\, v_1\rangle$, with $\langle k\, \Box\rangle\ \mathcal{R}^{\mathsf{C}}\ \langle k\, \Box\rangle$, and $v_0\ \mathcal{R}^{\mathsf{NF}\eta}\ v_1$, as wished.

If $t_0 \Downarrow_{\mathrm{v}} F_0[x\, v_0]$, then by definition of $\mathbb{M}$, there exist F_1, v_1, such that $t_1 \Downarrow_{\mathrm{v}} F_1[x\, v_1]$, $F_0\ \mathbb{M}^{\mathsf{C}}\ F_1$, and $v_0\ \mathbb{M}^{\mathsf{NF}\eta}\ v_1$. Because F_0 and F_1 are delimited (they are resulting from the evaluation of delimited terms), we have $F_0 = F_0'[\langle E_0\rangle]$, $F_1 = F_1'[\langle E_1\rangle]$, and $F_0\ \mathbb{M}^{\mathsf{C}}\ F_1$ implies $F_0'[y]\ \mathbb{M}\ F_1'[y]$ and $\langle E_0[y]\rangle\ \mathbb{M}\ \langle E_1[y]\rangle$ for any fresh y. As a result, we have we have $\langle k\, t_0\rangle \Downarrow_{\mathrm{v}} \langle k\, F_0'[\langle E_0[x\, v_0]\rangle]\rangle$, $\langle k\, t_1\rangle \Downarrow_{\mathrm{v}} \langle k\, F_1'[\langle E_1[x\, v_1]\rangle]\rangle$, and what remains to prove is that $\langle k\, F_0'[y]\rangle\ \mathcal{R}\ \langle k\, F_1'[y]\rangle$ holds for some fresh y. This is a consequence of $F_0'[y]\ \mathbb{M}\ F_1'[y]$, whether $F_0'[y]$ and $F_1'[y]$ are both delimited (then $\langle k\, F_0'[y]\rangle\ \mathcal{R}\ \langle k\, F_1'[y]\rangle$ holds by definition of $\mathcal{R}$) or not (then we have $\langle k\, F_0'[y]\rangle\ \mathbb{M}\ \langle k\, F_1'[y]\rangle$).

Lemma A.27. *If $x\ \mathbb{M}^{\mathsf{NF}\eta}\ v$, then for all fresh y, k, there exist F, E, and v' such that $\langle k\, (v \star y)\rangle \Downarrow_{\mathrm{v}} F[\langle E[x\, v']\rangle]$, $v'\ \mathbb{M}^{\mathsf{NF}\eta}\ y$, $\Box\ [\mathbb{M}]\ F$, and $\langle E\rangle\ \mathbb{M}^{\mathsf{C}}\ \langle k\, \Box\rangle$.*

Proof. Unfolding the definition of $x\ \mathbb{M}^{\mathsf{NF}\eta}\ v$, we know there exist F', v' such that $\langle k\, (v \star y)\rangle \Downarrow_{\mathrm{v}} \langle F'[x\, v']\rangle$, $v'\ \mathbb{M}^{\mathsf{NF}\eta}\ y$, and $\langle F'\rangle\ \mathbb{M}^{\mathsf{C}}\ \langle k\, \Box\rangle$ (we know there is a reset surrounding F' because $\langle k\, (v \star y)\rangle$ is delimited). From $\langle F'\rangle\ \mathbb{M}^{\mathsf{C}}\ \langle k\, \Box\rangle$, we deduce that $\langle F'\rangle = F[\langle E\rangle]$ with $F[z]\ \mathbb{M}\ z$ and $\langle E\rangle\ \mathbb{M}^{\mathsf{C}}\ \langle k\, \Box\rangle$. Either there is only one reset surrounding the hole in $\langle F'\rangle$, and $F = \Box$, $F' = E$, or there are more than one, and F is delimited. In both cases, we have $F\ [\mathbb{M}]\ \Box$, as required.

Lemma A.28. *If $y_0, \ldots, y_m$ is a sequence of variables such that $\overrightarrow{\sigma}(y_i) = v_{i+1}$ for $0 \leq i \leq m-1$ with $y_i\ \mathbb{M}^{\mathsf{NF}\eta}\ v_i$ for $1 \leq i \leq m$, then for any sequences of fresh variables $z_1, \ldots, z_m$, $k_1 \ldots k_m$, value v, and context E, there exist contexts $E_0, \ldots, E_m$, $F_1, \ldots, F_m$, and values $v_0', \ldots, v_m'$ such that*

$$\begin{aligned}&\langle E[y_0\, v]\rangle\overrightarrow{\sigma}\\ &\quad \to_{\mathrm{v}}^{*} F_1 \ldots F_m[\langle E_m[y_m\, v_m']\rangle]\{E_{m-1}/k_m\}\{v_{m-1}'/z_m\} \ldots \{E_0/k_1\}\{v_0'/z_1\}\overrightarrow{\sigma}\end{aligned}$$

where for $1 \leq i \leq m$, $\square$ $[\mathbb{M}]$ F_i, z_i $\mathbb{M}^{\mathsf{NF}\eta}$ v'_i, $\langle k_i \,\square\rangle$ $\mathbb{M}^{\mathsf{C}}$ $\langle E_i\rangle$ *and also* $v'_0 = v$, $E_0 = E$.

Proof. By induction on m, using Lemma A.27.

Lemma A.29. *If* $\overrightarrow{F} = F_1 \dots F_m$ *is a sequence of contexts such that for each* $1 \leq i \leq k$, $\square$ $[\mathbb{M}]$ F_i, *then for any fresh variables* $z_1, \dots, z_m$ *and value* v *there exist values* $v_0 \dots v_m$ *such that* $\overrightarrow{F}[v] \Downarrow_{\mathrm{v}} v_0\{v_1/z_1\}\dots\{v_m/z_m\}$ *with* z_i $\mathbb{M}^{\mathsf{NF}\eta}$ v_{i-1} *for* $1 \leq i \leq m$ *and* $v_m = v$.

Proof. By induction on m, using Lemma A.25.

Lemma A.30. *If* $k_0, \dots, k_m$ *is a sequence of variables such that* $\overrightarrow{\sigma}(k_i) = E_{i+1}$ *for* $0 \leq i \leq m-1$ *with* $\langle k_i \,\square\rangle$ $\mathbb{M}^{\mathsf{C}}$ $\langle E_i\rangle$ *for* $1 \leq i \leq m$, *then for any sequence of fresh variables* $x_1, \dots, x_m$, *and value* v, *there exist contexts* $F_1, \dots, F_m$, *and values* $v_0, \dots, v_m$ *such that*

$$\langle k_0\; v\rangle \overrightarrow{\sigma} \rightarrow_{\mathrm{v}}^{*} F_1 \dots F_m[\langle k_m\; v_m\rangle]\{v_{m-1}/x_m\}\dots\{v_0/x_1\}\overrightarrow{\sigma}$$

where for $1 \leq i \leq m$, $\square$ $[\mathbb{M}]$ F_i, x_i $\mathbb{M}^{\mathsf{NF}\eta}$ v_i, *and also* $v_0 = v$.

Proof. By induction on m.

Lemma A.31. *If* $\overrightarrow{F_0}$ $[\mathbb{M}]$ $\overrightarrow{F_1}$, $\overrightarrow{\sigma_0}$ $\langle\mathbb{M}\rangle$ $\overrightarrow{\sigma_1}$, *and* v_0 $\mathbb{M}^{\mathsf{NF}\eta}$ v_1, *then there exists* t'_1 *such that* $\overrightarrow{F_1}[v_1]\overrightarrow{\sigma_1} \rightarrow_{\mathrm{v}}^{*} t'_1$ *and* $\overrightarrow{F_1}[v_1]\overrightarrow{\sigma_1}$ $\mathcal{R}''$ t'_1.

Proof. Writing $\overrightarrow{F_0} = \overrightarrow{F'_0}\overrightarrow{\square}$, so that $\overrightarrow{F'_0}$ is either empty or the last element of $\overrightarrow{F'_0}$ is not $\square$, we have $\overrightarrow{F_1} = \overrightarrow{F'_1}\overrightarrow{F_1^{(4)}}$ with $\overrightarrow{F'_0}$ $[\mathbb{M}]$ $\overrightarrow{F'_1}$ and $\overrightarrow{\square}$ $[\mathbb{M}]$ $\overrightarrow{F_1^{(4)}}$. Therefore, by Lemma A.29, for any fresh variables $z_1, \dots, z_m$, there exist values $v'_0 \dots v'_m$ such that $\overrightarrow{F_1^{(4)}}[v_1] \Downarrow_{\mathrm{v}} v'_0\{v'_1/z_1\}\dots\{v'_m/z_m\}$ with z_i $\mathbb{M}^{\mathsf{NF}\eta}$ v'_{i-1} for $1 \leq i \leq m$ and $v'_m = v_1$. Then we have $\overrightarrow{F_1}[t_1]\overrightarrow{\sigma_1} \rightarrow_{\mathrm{v}}^{*} \overrightarrow{F'_1}[v'_0]\{v'_1/z_1\}\dots\{v'_m/z_m\}\overrightarrow{\sigma_1}$

If the sequences $\overrightarrow{F'_0}$ and $\overrightarrow{F'_1}$ are empty, then we have in fact $\overrightarrow{F_0}[v_0]\overrightarrow{\sigma_0} = v_0\overrightarrow{\sigma_0}$, and $\overrightarrow{F_1}[t_1]\overrightarrow{\sigma_1} \rightarrow_{\mathrm{v}}^{*} v'_0\{v'_1/z_1\}\dots\{v'_m/z_m\}\overrightarrow{\sigma_1}$, but $v_0\overrightarrow{\sigma_0}$ can be rewritten into $z_1\{z_2/z_1\}\dots\{v_0/z_m\}\overrightarrow{\sigma_0}$, so we obtain terms in (the second set of) $\mathcal{R}''$.

Otherwise, we have $\overrightarrow{F_0}[t_0]\overrightarrow{\sigma_0} \rightarrow_{\mathrm{v}} \overrightarrow{F_0^{(3)}}[F''_0[v_0]]\overrightarrow{\sigma_0}$, where F''_0 is the last context of the sequence $\overrightarrow{F'_0}$, and similarly $\overrightarrow{F_1}[t_1]\overrightarrow{\sigma_1} \rightarrow_{\mathrm{v}}^{*} \overrightarrow{F_1^{(3)}}[F''_1[v'_0]]\{v'_1/z_1\}\dots\{v'_m/z_m\}\overrightarrow{\sigma_1}$. But these two resulting terms can be rewritten into respectively

$$\overrightarrow{F_0^{(3)}}[F''_0[z]]\{z_1/z\}\{z_2/z_1\}\dots\{v_0/z_m\}\overrightarrow{\sigma_0} \tag{A.1}$$

$$\overrightarrow{F_1^{(3)}}[F''_1[z]]\{v'_0/z\}\{v'_1/z_1\}\dots\{v'_m/z_m\}\overrightarrow{\sigma_1} \tag{A.2}$$

for a fresh z, and since F''_0 is not $\square$ and also F''_0 $[\mathbb{M}]$ F''_1, we have $F''_0[z]$ $\mathbb{M}$ $F''_1[z]$ and these two terms are delimited. Consequently, the terms A.1 and A.2 are related by $\mathcal{R}''$.

Lemma A.32. *Let* $\overrightarrow{F_0}$ $[\mathbb{M}]$ $\overrightarrow{F_1}$*,* $\overrightarrow{\sigma_0}$ $\langle\mathbb{M}\rangle$ $\overrightarrow{\sigma_1}$*,* $\langle E'_0\rangle$ $\mathbb{M}^{\mathsf{C}}$ $\langle E'_1\rangle$*.*

- *If* $x\overrightarrow{\sigma_0} = y$*, then there exists* t_1 *such that* $\overrightarrow{F_1}[\langle E'_1[x\, v_1]\rangle]\overrightarrow{\sigma_1} \Downarrow_{\mathrm{v}} t_1$ *and* $\overrightarrow{F_0}[\langle E'_0[x\, v_0]\rangle]\overrightarrow{\sigma_0}\ \mathcal{R}''^{\mathsf{NF}\eta}\ t_1$*.*
- *If* $x\overrightarrow{\sigma_0}$ *is a* λ*-abstraction, then there exist* t_0*,* t_1 *such that* $\overrightarrow{F_0}[\langle E'_0[x\, v_0]\rangle]\overrightarrow{\sigma_0} \rightarrow_{\mathrm{v}} t_0$*,* $\overrightarrow{F_1}[\langle E'_1[x\, v_1]\rangle]\overrightarrow{\sigma_1} \rightarrow_{\mathrm{v}}^* t_1$*, and* $t_0\ \mathcal{R}''\ t_1$*.*

Proof. There is a sequence of variables $y_0 \dots y_m$ such that $y_0 = x$, $\overrightarrow{\sigma_0}(y_i) = y_{i+1}$ for $0 \leq i \leq m-1$, $\overrightarrow{\sigma_0}(y_m) = v$, v is a λ-abstraction or v is a variable not in the domain of $\overrightarrow{\sigma_0}$. For $0 \leq i \leq m$, we write $\overrightarrow{\sigma_1}(y_i) = v_1^i$, and we have $y_i\ \mathbb{M}^{\mathsf{NF}\eta}\ v_1^{i-1}$ for $1 \leq i \leq m$, and $v\ \mathbb{M}^{\mathsf{NF}\eta}\ v_1^m$. Let $z_1, \dots, z_m$, $k_1 \dots k_m$ be fresh variables. By Lemma A.28, there exist contexts $E_1^0, \dots, E_1^m$, $\overrightarrow{F_1''} = F_1''^1, \dots, F_1''^m$, and values ${v'_1}^0, \dots, {v'_1}^m$ such that

$$\begin{aligned}&\overrightarrow{F_1}[\langle E'_1[x\, v_1]\rangle]\overrightarrow{\sigma_1} \rightarrow_{\mathrm{v}}^*\\ &\qquad \overrightarrow{F_1}[\overrightarrow{F_1''}[\langle E_1^m[y_m\, {v'_1}^m]\rangle]]\{E_1^{m-1}/k_m\}\{{v'_1}^{m-1}/z_m\}\dots\{E_1^0/k_1\}\{{v'_1}^0/z_1\}\overrightarrow{\sigma_1}\end{aligned}$$

where for $1 \leq i \leq m$, $\square\ [\mathbb{M}]\ F_1''^i$, $z_i\ \mathbb{M}^{\mathsf{NF}\eta}\ {v'_1}^i$, $\langle k_i\, \square\rangle\ \mathbb{M}^{\mathsf{C}}\ \langle E_1^i\rangle$ and also ${v'_1}^0 = v_1$, $E_1^0 = E'_1$. We now distinguish cases based on v.

If v is a variable y, then $\overrightarrow{F_0}[\langle E'_0[x\, v_0]\rangle]\overrightarrow{\sigma_0}$ is an open stuck term. Let z_{m+1}, k_{m+1} be fresh variables. From $y\ \mathbb{M}^{\mathsf{NF}\eta}\ v_1^m$ and Lemma A.27, we know there exist contexts $F_1''^{m+1}$, E_1^{m+1}, and a value ${v'_1}^{m+1}$ such that $\langle k_{m+1}\, (v_1^m \star z_{m+1})\rangle \Downarrow_{\mathrm{v}} F_1''^{m+1}[\langle E_1^{m+1}[y\, {v'_1}^{m+1}]\rangle]$, $z_{m+1}\ \mathbb{M}^{\mathsf{NF}\eta}\ {v'_1}^{m+1}$, $\square\ [\mathbb{M}]\ F_1''^{m+1}$, $\langle k_{m+1}\, \square\rangle\ \mathbb{M}^{\mathsf{C}}\ \langle E_1^{m+1}\rangle$. As a result, we have

$$\begin{aligned}&\overrightarrow{F_1}[\langle E'_1[x\, v_1]\rangle]\overrightarrow{\sigma_1} \rightarrow_{\mathrm{v}}^*\\ &\qquad \overrightarrow{F_1}[\overrightarrow{F_1''}[F_1''^{m+1}[\langle E_1^{m+1}[y\, {v'_1}^{m+1}]\rangle]]]\{E_1^m/k_{m+1}\}\{{v'_1}^m/z_{m+1}\}\\ &\qquad\qquad \{E_1^{m-1}/k_m\}\{{v'_1}^{m-1}/z_m\}\dots\{E_1^0/k_1\}\{{v'_1}^0/z_1\}\overrightarrow{\sigma_1}\end{aligned}$$

We can also write $\overrightarrow{F_0}[\langle E'_0[x\, v_0]\rangle]\overrightarrow{\sigma_0}$ as

$$\begin{aligned}&\overrightarrow{F_0}[\overrightarrow{\square}[\langle k_{m+1}\, (y\, z_{m+1})\rangle]]\{k_m\, \square/k_{m+1}\}\{z_m/z_{m+1}\}\\ &\qquad \{k_{m-1}\, \square/k_m\}\{z_{m-1}/z_m\}\dots\{E'_0/k_1\}\{v_0/z_1\}\overrightarrow{\sigma_0}\end{aligned}$$

The resulting open stuck terms are related by $\mathcal{R}''$, as wished.

If v is a λ-abstraction $\lambda y.t$, then $\overrightarrow{F_0}[\langle E'_0[x\, v_0]\rangle]\overrightarrow{\sigma_0}$ is able to reduce. From $\lambda y.t\ \mathbb{M}^{\mathsf{NF}\eta}\ v_1^m$, we deduce $t\ \mathbb{M}\ v_1^m \star y$ (assuming y fresh enough, which is always possible with α-conversion), which in turn implies $\langle k_{m+1}\, t\rangle\ \mathbb{M}\ \langle k_{m+1}\, (v_1^m \star y)\rangle$ for any fresh k_{m+1} (using Lemma A.26). As a result, we have

$$\begin{aligned}&\overrightarrow{F_1}[\langle E'_1[x\, v_1]\rangle]\overrightarrow{\sigma_1} \rightarrow_{\mathrm{v}}^* \overrightarrow{F_1}[\overrightarrow{F_1''}[\langle k_{m+1}\, (v_1^m \star y)\rangle]]\{E_1^m/k_{m+1}\}\{{v'_1}^m/y\}\\ &\qquad\qquad \{E_1^{m-1}/k_m\}\{{v'_1}^{m-1}/z_m\}\dots\{E_1^0/k_1\}\{{v'_1}^0/z_1\}\overrightarrow{\sigma_1}\end{aligned}$$

and we also have

$$\overrightarrow{F_0}[\langle E_0'[x\, v_0]\rangle]\overrightarrow{\sigma_0} \rightarrow_{\mathrm{v}} \overrightarrow{F_0}[\overrightarrow{\Box}[\langle k_{m+1}\, t\rangle]]\{k_m\, \Box/k_{m+1}\}\{z_m/y\}$$
$$\{k_{m-1}\, \Box/k_m\}\{z_{m-1}/z_m\}\ldots\{E_0'/k_1\}\{v_0/z_1\}\overrightarrow{\sigma_0}$$

The two resulting terms are related by $\mathcal{R}''$, as required.

Lemma A.33. *$\mathcal{R}''$ is a delimited normal-form bisimulation.*

Proof. By case analysis on the terms in $\mathcal{R}''$. For terms of the form $v_0\overrightarrow{\sigma_0}$, $v_1\overrightarrow{\sigma_1}$, we have to prove that $\langle k\, v_0\overrightarrow{\sigma_0}\rangle\ \mathcal{R}''\ \langle k\, v_1\overrightarrow{\sigma_1}\rangle$ holds for a fresh k. By definition, we have $v_0\ \mathbb{M}\ v_1$, which implies $\langle k\, v_0\rangle\ \mathbb{M}\ \langle k\, v_1\rangle$. We obtain delimited terms, therefore we have $\langle k\, v_0\rangle\overrightarrow{\sigma_0}\ \mathcal{R}''\ \langle k\, v_1\rangle\overrightarrow{\sigma_1}$, which is the same as $\langle k\, v_0\overrightarrow{\sigma_0}\rangle\ \mathcal{R}''\ \langle k\, v_1\overrightarrow{\sigma_1}\rangle$, as wished.

We now consider terms of the form $\overrightarrow{F_0}[t_0]\overrightarrow{\sigma_0}$, $\overrightarrow{F_1}[t_1]\overrightarrow{\sigma_1}$. Note that by definition of $\mathcal{R}''$ and $[\mathbb{M}]$, these terms are delimited. We distinguish cases based on t_0.

Case: $t_0 \rightarrow_{\mathrm{v}} t_0'$, with t_0' delimited. Then $\overrightarrow{F_0}[t_0]\overrightarrow{\sigma_0} \rightarrow_{\mathrm{v}} \overrightarrow{F_0}[t_0']\overrightarrow{\sigma_0}$. Because $\rightarrow_{\mathrm{v}} \subseteq \mathbb{M}$, we have $\overrightarrow{F_0}[t_0']\overrightarrow{\sigma_0}\ \mathcal{R}''\ \overrightarrow{F_1}[t_1]\overrightarrow{\sigma_1}$, as required.

Case: $t_0 \rightarrow_{\mathrm{v}} v_0$. Then $\overrightarrow{F_0}[t_0]\overrightarrow{\sigma_0} \rightarrow_{\mathrm{v}} \overrightarrow{F_0}[v_0]\overrightarrow{\sigma_0}$. By bisimilarity, there exists v_1 such that $t_1 \Downarrow_{\mathrm{v}} v_1$ and $v_0\ \mathbb{M}^{\mathsf{NF}\eta}\ v_1$. We can conclude by Lemma A.31.

Case: $t_0 = \langle F_0[x\, v_0]\rangle$. Since $t_0\ \mathbb{M}\ t_1$, there exist F_1, v_1 such that $t_1 \Downarrow_{\mathrm{v}} F_1[x\, v_1]$, $\langle F_0\rangle\ \mathbb{M}^{\mathsf{C}}\ F_1$ and $v_0\ \mathbb{M}^{\mathsf{NF}\eta}\ v_1$. We know that F_1 is delimited, as (part of) the result of the evaluation of the delimited term t_1. The relation $\langle F_0\rangle\ \mathbb{M}^{\mathsf{C}}\ F_1$ implies that $\langle F_0\rangle = F_0'[\langle E_0'\rangle]$, $F_1 = F_1'[\langle E_1'\rangle]$, $F_0'\ [\mathbb{M}]\ F_1'$, and $\langle E_0'\rangle\ \mathbb{M}^{\mathsf{C}}\ \langle E_1'\rangle$. We can conclude by Lemma A.32.

Case: $t_0 = F_0'[\langle k\, v_0\rangle]$. Since $t_0\ \mathbb{M}\ t_1$, there exists F_1', v_1, such that $t_1 \Downarrow_{\mathrm{v}} F_1'[\langle k\, v_1\rangle]$, $F_0'[\langle\Box\rangle]\ \mathbb{M}^{\mathsf{C}}\ F_1'[\langle\Box\rangle]$ and $v_0\ \mathbb{M}^{\mathsf{NF}\eta}\ v_1$. Note that we know there is a reset surrounding $k\, v_1$ because t_1 is well formed w.r.t. k and by applying Proposition 4.22. Because t_0, t_1 are delimited, and $F_0'[\langle\Box\rangle]\ \mathbb{M}^{\mathsf{C}}\ F_1'[\langle\Box\rangle]$, we also have $F_0'\ [\mathbb{M}]\ F_1'$.

There is a sequence of variables $k_0 \ldots k_m$ such that $k_0 = k$, $\overrightarrow{\sigma_0}(k_i) = k_{i+1}\, \Box$ for $0 \leq i \leq m-1$ and $\overrightarrow{\sigma_0}(k_i) = E$ so that if $E = k\, \Box$, k is not in the domain of $\overrightarrow{\sigma_0}$. For $0 \leq i \leq m$, we write $\overrightarrow{\sigma_1}(k_i) = E_1^i$, and we have $\langle k_i\, \Box\rangle\ \mathbb{M}^{\mathsf{C}}\ \langle E_1^{i-1}\rangle$ for $1 \leq i \leq m$, and $\langle E\rangle\ \mathbb{M}^{\mathsf{C}}\ \langle E_1^m\rangle$. Let $x_1 \ldots x_m$ be fresh variables. By Lemma A.30, there exist $\overrightarrow{F_1''} = {F_1''}^1 \ldots {F_1''}^m$, $v_1^0 \ldots v_1^m$ such that

$$\overrightarrow{F_1}[t_1]\overrightarrow{\sigma_1} \rightarrow_{\mathrm{v}}^* \overrightarrow{F_1}[F_1'[\overrightarrow{F_1''}[\langle k_m\, v_1^m\rangle]]]\{v_1^{m-1}/x_m\}\ldots\{v_1^1/x_2\}\{v_1^0/x_1\}\overrightarrow{\sigma_1}$$

with for $1 \leq i \leq m$, $\Box\ [\mathbb{M}]\ {F_1''}^i$, $x_i\ \mathbb{M}^{\mathsf{NF}\eta}\ v_1^i$, and $v_1^0 = v_1$. We distinguish cases based on $\langle E[x]\rangle$, where x is a fresh variable.

If $\langle E[x]\rangle \rightarrow_{\mathrm{v}} t_0'$, with t_0' delimited, then because $\rightarrow_{\mathrm{v}} \subseteq \mathbb{M}$ and $\langle E\rangle\ \mathbb{M}^{\mathsf{C}}\ \langle E_1^m\rangle$, we have $t_0'\ \mathbb{M}\ \langle E_1^m[x]\rangle$. On one hand, we have

$$\overrightarrow{F_1}[t_1]\overrightarrow{\sigma_1} \rightarrow_{\mathrm{v}}^* \overrightarrow{F_1}[F_1'[\overrightarrow{F_1''}[\langle E_1^m[x]\rangle]]]\{v_1^m/x\}\{v_1^{m-1}/x_m\}\ldots\{v_1^1/x_2\}\{v_1^0/x_1\}\overrightarrow{\sigma_1}$$

and on the other hand, we have

$$\overrightarrow{F_0}[t_0]\overrightarrow{\sigma_0} \rightarrow_{\mathrm{v}} \overrightarrow{F_1}[F_0'[\overrightarrow{\Box}[\langle t_0'\rangle]]]\{x_m/x\}\{x_{m-1}/x_m\}\ldots\{x_1/x_2\}\{v_0/x_1\}\overrightarrow{\sigma_0}$$

The two resulting terms are related by $\mathcal{R}''$, as wished.

If $\langle E[x]\rangle \rightarrow_{\mathrm{v}} v$, then in fact $E = \Box$ and $v = x$. Because $\langle x\rangle\ \mathbb{M}\ \langle E_1^m[x]\rangle$, by bisimilarity, there exists v' such that $\langle E_1^m[x]\rangle \Downarrow_{\mathrm{v}} v'$ and $x\ \mathbb{M}^{\mathsf{NF}\eta}\ v'$. Consequently, we have

$$\overrightarrow{F_1}[t_1]\overrightarrow{\sigma_1} \rightarrow_{\mathrm{v}}^* \overrightarrow{F_1}[F_1'[\overrightarrow{F_1''}[v']]]\{v_1^m/x\}\{v_1^{m-1}/x_m\}\ldots\{v_1^1/x_2\}\{v_1^0/x_1\}\overrightarrow{\sigma_1}$$

but we also have

$$\overrightarrow{F_0}[t_0]\overrightarrow{\sigma_0} \rightarrow_{\mathrm{v}} \overrightarrow{F_1}[F_0'[\overrightarrow{\Box}[x]]]\{x_m/x\}\{x_{m-1}/x_m\}\ldots\{x_1/x_2\}\{v_0/x_1\}\overrightarrow{\sigma_0}$$

From there, we can conclude using Lemma A.31.

If $\langle E[x]\rangle$ is an open stuck term of the form $F_0''[y\ v_0']$, then since $\langle E[x]\rangle\ \mathbb{M}\ \langle E_1^m[x]\rangle$, there exist $F_1^{(4)}$, v_1' such that $\langle E_1^m[x]\rangle \Downarrow_{\mathrm{v}} F_1^{(4)}[y\ v_1']$, $F_0''\ \mathbb{M}^{\mathsf{C}}\ F_1^{(4)}$, and $v_0'\ \mathbb{M}^{\mathsf{NF}\eta}\ v_1'$. The context F_0'' is delimited, therefore it can be written $F_0'' = F_0^{(3)}[\langle E_0'\rangle]$. From $F_0''\ \mathbb{M}^{\mathsf{C}}\ F_1^{(4)}$, we know there exists $F_1^{(3)}$, E_1' such that $F_0^{(3)}\ [\mathbb{M}]\ F_1^{(3)}$ and $\langle E_0'[z]\rangle\ \mathbb{M}\ \langle E_1'[z]\rangle$ for a fresh z. As a result, we have

$$\begin{aligned}&\overrightarrow{F_1}[t_1]\overrightarrow{\sigma_1} \rightarrow_{\mathrm{v}}^* \\ &\qquad \overrightarrow{F_1}[F_1'[\overrightarrow{F_1''}[F_1^{(3)}[\langle E_1'[y\ v_1']\rangle]]]]\{v_1^m/x\}\{v_1^{m-1}/x_m\}\ldots\{v_1^1/x_2\}\{v_1^0/x_1\}\overrightarrow{\sigma_1}\end{aligned}$$

but we also have

$$\overrightarrow{F_0}[t_0]\overrightarrow{\sigma_0} = \overrightarrow{F_1}[F_0'[\overrightarrow{\Box}[F_0^{(3)}[\langle E_0'[y\ v_0']\rangle]]]]\{x_m/x\}\{x_{m-1}/x_m\}\ldots\{x_1/x_2\}\{v_0/x_1\}\overrightarrow{\sigma_0}$$

We can therefore conclude with Lemma A.32.

If $\langle E[x]\rangle = \langle k'\ x\rangle$, (i.e., $E = k'\ \Box$), then since $\langle E[x]\rangle\ \mathbb{M}\ \langle E_1^m[x]\rangle$, there exist $F_1^{(3)}$, v_1'' such that $\langle E_1^m[x]\rangle \Downarrow_{\mathrm{v}} F_1^{(3)}[\langle k'\ v_1''\rangle]$, $\Box\ [\mathbb{M}]\ F_1^{(3)}$, and $x\ \mathbb{M}^{\mathsf{NF}\eta}\ v_1''$. Consequently, we have

$$\begin{aligned}&\overrightarrow{F_1}[t_1]\overrightarrow{\sigma_1} \rightarrow_{\mathrm{v}}^* \\ &\qquad \overrightarrow{F_1}[F_1'[\overrightarrow{F_1''}[F_1^{(3)}[\langle k'\ v_1''\rangle]]]]\{v_1^m/x\}\{v_1^{m-1}/x_m\}\ldots\{v_1^1/x_2\}\{v_1^0/x_1\}\overrightarrow{\sigma_1}\end{aligned}$$

and also

$$\overrightarrow{F_0}[t_0]\overrightarrow{\sigma_0} = \overrightarrow{F_1}[F_0'[\overrightarrow{\Box}[\langle k'\ x\rangle]]]\{x_m/x\}\{x_{m-1}/x_m\}\ldots\{x_1/x_2\}\{v_0/x_1\}\overrightarrow{\sigma_0}$$

We obtain two open stuck terms related by $\mathcal{R}''^{\mathsf{NF}\eta}$, as required.

Theorem A.34. *The relation* $\mathbb{M}$ *is a congruence.*

Proof. Let $t_0 \mathrel{\mathbb{M}} t_1$. We first prove compatibility w.r.t. application and reset. For a fresh k, we have $\langle k\, t_0\rangle \mathrel{\mathbb{M}} \langle k\, t_1\rangle$ by Lemma A.26, which in turn implies $\langle k\, t_0\rangle\sigma_0 \mathrel{\mathbb{M}} \langle k\, t_1\rangle\sigma_1$ for $\sigma_0 \mathrel{\langle\mathbb{M}\rangle} \sigma_1$ according to Lemma A.33. By taking $\sigma_0 = \sigma_1 = \{\square/k\}$, we obtain $\langle t_0\rangle \mathrel{\mathbb{M}} \langle t_1\rangle$, which gives us compatibility w.r.t. reset. By taking $\sigma_0 = \sigma_1 = \{k'\ (\square\ t)/k\}$ where k' fresh, we obtain $\langle k'\ (t_0\ t)\rangle \mathrel{\mathbb{M}} \langle k'\ (t_1\ t)\rangle$, which implies $t_0\, t \mathrel{\mathbb{M}} t_1\, t$.

To prove that we also have $t\, t_0 \mathrel{\mathbb{M}} t\, t_1$, let k', x be fresh variables. By taking $\sigma_0 = \sigma_1 = \{k'\ (x\ \square)/k\}$, we obtain $\langle k'\,(x\, t_0)\rangle \mathrel{\mathbb{M}} \langle k'\,(x\, t_1)\rangle$, which implies $\{k'\ (\square\ t_0)/k''\} \mathrel{[\mathbb{M}]} \{k'\ (\square\ t_1)/k''\}$ for all k''. We therefore have $\langle k''\, t\rangle\{k'\ (\square\ t_0)/k''\} \mathrel{\mathbb{M}} \langle k''\, t\rangle\{k'\ (\square\ t_1)/k''\}$ (k'' fresh), i.e., $\langle k'\,(t\, t_0)\rangle \mathrel{\mathbb{M}} \langle k'\,(t\, t_1)\rangle$, which means that $t\, t_0 \mathrel{\mathbb{M}} t\, t_1$ holds, as wished.

We now prove compatibility w.r.t. λ-abstraction. From $t_0 \mathrel{\mathbb{M}} t_1$, we obtain $\lambda x.t_0 \mathrel{\mathbb{M}^{\mathsf{NF}\eta}} \lambda x.t_1$, which in turn implies $\langle k\ \lambda x.t_0\rangle \mathrel{\mathbb{M}} \langle k\ \lambda x.t_0\rangle$ (k fresh), which finally gives us $\lambda x.t_0 \mathrel{\mathbb{M}} \lambda x.t_1$, as required.

Finally, we prove compatibility w.r.t. shift. Let k' be a fresh variable. From $t_0 \mathrel{\mathbb{M}} t_1$, we obtain $\langle t_0\{\lambda x.\langle k'\ x\rangle/k\}\rangle \mathrel{\mathbb{M}} \langle t_1\{\lambda x.\langle k'\ x\rangle/k\}\rangle$ (x fresh). One can then prove that $\{(\mathcal{S}k.t_0, \mathcal{S}k.t_1), (\langle k'\ \mathcal{S}k.t_0\rangle, \langle k'\ \mathcal{S}k.t_1\rangle) \mid k' \notin \mathsf{fv}(t_0) \cup \mathsf{fv}(t_1)\} \cup \mathbb{M}$ is a delimited normal-form bisimulation.

A.5. Delimited bisimulation up to context

Lemma A.35. *If* $\mathcal{R}$ *is a delimited bisimulation up to context,* $v_0 \mathrel{\breve{\mathcal{R}}} t_1$*, and* t_1 *is delimited, then there exists* v_1 *such that* $t_1 \Downarrow_{\mathrm{v}} v_1$ *and* $v_0 \mathrel{\breve{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1$*.*

Proof. By induction on the derivation of $v_0 \mathrel{\breve{\mathcal{R}}} t_1$. If $v_0 = t_1$, then the result is direct. The result is also direct in the case of compatibility w.r.t. λ-abstraction.

If $v_0 \mathrel{\mathcal{R}} t_1$, then $\langle k\, v_0\rangle \mathrel{\mathcal{R}} \langle k\, t_1\rangle$ holds for a fresh k, and there exists t_1' such that $\langle k\, t_1\rangle \Downarrow_{\mathrm{v}} t_1'$ and $\langle k\, v_0\rangle \mathrel{\breve{\mathcal{R}}^{\mathsf{NF}\searrow}} t_1'$. But t_1 is delimited, so the only possibility is $t_1' = \langle k\, v_1\rangle$ for some v_1 with $t_1 \Downarrow_{\mathrm{v}} v_1$ and $v_0 \mathrel{\breve{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1$, hence the result holds.

The cases $v_0 = v_0'\{v_0''/x\}$, $t_1 = t_1'\{v_1''/x\}$ with $v_0' \mathrel{\breve{\mathcal{R}}} t_1'$, $v_0 \mathrel{\breve{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1$, and $v_0 = v_0'\{E_0/k\}$, $t_1 = t_1'\{E_1/k\}$ with $v_0' \mathrel{\breve{\mathcal{R}}} t_1'$, $\langle E_0\rangle \mathrel{\breve{\mathcal{R}}^{\mathsf{C}\searrow}} \langle E_1\rangle$ are straightforward by induction.

If $t_1 = E_1[t_1']$ with $v_0 \mathrel{\breve{\mathcal{R}}} t_1'$ and $x \mathrel{\breve{\mathcal{R}}} E_1[x]$, then in fact $E_1 = \square$ and t_1' is delimited, otherwise t_1 would not be delimited. Therefore, by the induction hypothesis, there exists v_1 such that $t_1' \Downarrow_{\mathrm{v}} v_1$ with $v_0 \mathrel{\breve{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1$. As a result, we have $t_1 \Downarrow_{\mathrm{v}} v_1$ with $v_0 \mathrel{\breve{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1$, as wished.

Lemma A.36. *If* $\mathcal{R}$ *is a bisimulation up to context,* $F_0[x] \mathrel{\breve{\mathcal{R}}^{\searrow}} F_1[x]$ *for a fresh* x*,* $t_0 \mathrel{\breve{\mathcal{R}}^{\mathsf{NF}\searrow}} t_1$*, and* t_0*,* t_1 *are delimited open stuck terms, then* $F_0[t_0] \mathrel{\breve{\mathcal{R}}^{\mathsf{NF}\searrow}} F_1[t_1]$*.*

Proof. Same as for Lemma A.19, except we use Lemma A.35.

Lemma A.37. *Let* $\mathcal{R}$ *be a delimited bisimulation up to context. Let* $t_0 \mathrel{\breve{\mathcal{R}}} t_1$*.*

1. *If t_0, t_1 are not both delimited, then for all fresh k, if $\langle k\, t_0\rangle \Downarrow_{\mathrm{v}} t'_0$, there exists t'_1 such that $\langle k\, t_1\rangle \Downarrow_{\mathrm{v}} t'_1$ and $t'_0 \mathrel{\breve{\mathcal{R}}^{\mathsf{NF}\searrow}} t'_1$ (and conversely if $\langle k\, t_1\rangle \Downarrow_{\mathrm{v}} t'_1$) .*
2. *If t_0, t_1 are both delimited and $t_0 \Downarrow_{\mathrm{v}} t'_0$, there exists t'_1 such that $t_1 \Downarrow_{\mathrm{v}} t'_1$ and $t'_0 \mathrel{\breve{\mathcal{R}}^{\mathsf{NF}\searrow}} t'_1$ (and conversely if $t_1 \Downarrow_{\mathrm{v}} t'_1$).*

Proof. As in the proof of Lemma A.21, we proceed by induction on the number of steps m it takes for t_0 or $\langle k\, t_0\rangle$ to reduce to a normal form, and on the derivation of $t_0 \mathrel{\breve{\mathcal{R}}} t_1$, ordered lexicographically. We only detail the differences with the original proof of Lemma A.21.

We suppose first that $m = 0$. The cases $t_0 \mathrel{\mathcal{R}} t_1$ and $t_0 = t_1$ are immediate. We detail the other cases.

Case: $t_0\{v_0/x\} \mathrel{\breve{\mathcal{R}}} t_1\{v_1/x\}$ with $t_0 \mathrel{\breve{\mathcal{R}}} t_1$ and $v_0 \mathrel{\breve{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1$. The terms $t_0\{v_0/x\}$ or $\langle k\, t_0\{v_0/x\}\rangle$ (k fresh) are normal forms if t_0 or $\langle k\, t_0\rangle$ are themselves normal forms. We can conclude by using the induction hypothesis on $t_0 \mathrel{\breve{\mathcal{R}}} t_1$, and then apply the substitutions on the resulting normal forms.

Case: $t_0\{E_0/k\} \mathrel{\breve{\mathcal{R}}} t_1\{E_1/k\}$ with $t_0 \mathrel{\breve{\mathcal{R}}} t_1$ and $E_0[x] \mathrel{\breve{\mathcal{R}}} E_1[x]$ (x fresh). Similar to the previous case.

Case: $\lambda x.t_0 \mathrel{\breve{\mathcal{R}}} \lambda x.t'_1$ with $t_0 \mathrel{\breve{\mathcal{R}}} t_1$. We have to prove $\langle k\, \lambda x.t_0\rangle \mathrel{\breve{\mathcal{R}}^{\mathsf{NF}\searrow}} \langle k\, \lambda x.t_1\rangle$, i.e., $\lambda x.t_0 \mathrel{\breve{\mathcal{R}}^{\mathsf{NF}\searrow}} \lambda x.t_1$, which is true since $t_0 \mathrel{\breve{\mathcal{R}}} t_1$.

Case: $F_0[t_0] \mathrel{\breve{\mathcal{R}}} F_1[t_1]$ with $F_0 \mathrel{\breve{\mathcal{R}}^{\mathsf{C}\searrow}} F_1$ and $t_0 \mathrel{\breve{\mathcal{R}}} t_1$. The term $F_0[t_0]$ is a normal form if t_0 is value v_0 and $F_0 = \square$ or if t_0 is an open stuck term. For the former case, we know by induction that for any fresh k, there exist F'_1, v_1 such that $\langle k\, t_1\rangle \Downarrow_{\mathrm{v}} F'_1[\langle k\, v_1\rangle]$, $\langle\square\rangle \mathrel{\breve{\mathcal{R}}^{\mathsf{C}\searrow}} F'_1[\langle\square\rangle]$, and $v_0 \mathrel{\breve{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1$. Let k' be a fresh variable. From $\square \mathrel{\breve{\mathcal{R}}^{\mathsf{C}\searrow}} F_1$, we deduce that F_1 is a pure context E_1, and for a fresh x, we have $\langle k'\, E_1[x]\rangle \Downarrow_{\mathrm{v}} F''_1[\langle k'\, x\rangle]$ with $\langle\square\rangle \mathrel{\breve{\mathcal{R}}^{\mathsf{C}\searrow}} F''_1[\langle\square\rangle]$ (by using the induction hypothesis). Consequently we have the transitions

$$\begin{aligned}\langle k'\, E_1[t_1]\rangle &\to_{\mathrm{v}}^{*} F'_1\{k'\ E_1/k\}[\langle k'\, E_1[v_1\{k'\ E_1/k\}]\rangle]\\ &\to_{\mathrm{v}}^{*} F'_1\{k'\ E_1/k\}[F''_1\{v_1\{k'\ E_1/k\}/x\}[\langle k'\, v_1\{k'\ E_1/k\}\rangle]]\end{aligned}$$

From $v_0 \mathrel{\breve{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1$, we deduce $v_0 \mathrel{\breve{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1\{k'\ E_1/k\}$ (because k is fresh w.r.t. v_0). From $\langle\square\rangle \mathrel{\breve{\mathcal{R}}^{\mathsf{C}\searrow}} F'_1[\langle\square\rangle]$ and $\langle\square\rangle \mathrel{\breve{\mathcal{R}}^{\mathsf{C}\searrow}} F''_1[\langle\square\rangle]$, we get $z \mathrel{\breve{\mathcal{R}}^{\searrow}} F'_1\{k'\ E_1/k\}[z]$ and $z \mathrel{\breve{\mathcal{R}}^{\searrow}} F''_1\{v_1\{k'\ E_1/k\}/x\}[z]$ for a fresh z. The latter and Lemma A.35 imply that $F''_1\{v_1\{k'\ E_1/k\}/x\}[z] \Downarrow_{\mathrm{v}} z$, from which we can deduce $z \mathrel{\breve{\mathcal{R}}^{\searrow}} F'_1\{k'\ E_1/k\}[F''_1\{v_1\{k'\ E_1/k\}/x\}[z]]$. Consequently, we obtain $\langle k'\, v_0\rangle \mathrel{\breve{\mathcal{R}}^{\mathsf{NF}\searrow}} F'_1\{k'\ E_1/k\}[F''_1\{v_1\{k'\ E_1/k\}/x\}[\langle k'\, v_1\{k'\ E_1/k\}\rangle]]$, as required.

We now suppose t_0 is an open stuck term. If both t_0 and t_1 are delimited, then by the induction hypothesis, there exists an open stuck term t'_1 such that

$t_1 \Downarrow_{\text{v}} t'_1$ and $t_0 \breve{\mathcal{R}}^{\mathsf{NF}\searrow} t'_1$. It is then easy to prove that $F_0[t_0] \breve{\mathcal{R}}^{\mathsf{NF}\searrow} F_1[t'_1]$ or $\langle k\, F_0[t_0]\rangle \breve{\mathcal{R}}^{\mathsf{NF}\searrow} \langle k\, F_1[t'_1]\rangle$, depending on whether F_0 and F_1 are both delimited or not. Otherwise, for a fresh k, there exists t'_1 such that $\langle k\, t_1\rangle \Downarrow_{\text{v}} t'_1$ and $\langle k\, t_0\rangle \breve{\mathcal{R}}^{\mathsf{NF}\searrow} t'_1$. If $F_0 = E_0$, then $F_1 = E_1$, and for any fresh k' and x, we have $\langle k'\, E_0[x]\rangle \breve{\mathcal{R}} \langle k'\, E_1[x]\rangle$, and also $\langle k'\, E_1[t_1]\rangle \Downarrow_{\text{v}} t'_1\{k'\ E_1/k\}$. We therefore have $\langle k\, t_0\rangle\{k'\ E_0/k\} \breve{\mathcal{R}}^{\mathsf{NF}\searrow} t'_1\{k'\ E_1/k\}$, i.e., $\langle k'\, E_0[t_0]\rangle \breve{\mathcal{R}}^{\mathsf{NF}\searrow} t'_1\{k'\ E_1/k\}$, hence the result holds. If $F_0 = F'_0[\langle E_0\rangle]$, then $F_1 = F'_1[\langle E_1\rangle]$ with $F'_0[z] \breve{\mathcal{R}}^{\searrow} F'_1[z]$ and $\langle E_0[z]\rangle \breve{\mathcal{R}} \langle E_1[z]\rangle$ for a fresh z. We assume F'_0, F'_1 are both delimited, the case where they are not is similar (just replace F'_0, F'_1 with $\langle k'\, F'_0\rangle$, $\langle k'\, F'_1\rangle$ for a fresh k'). The evaluation $\langle k\, t_1\rangle \Downarrow_{\text{v}} t'_1$ implies $F_1[t_1] \Downarrow_{\text{v}} F'_1[t'_1\{E_1/k\}]$, and $\langle k\, t_0\rangle \breve{\mathcal{R}}^{\mathsf{NF}\searrow} t'_1$ implies $F'_0[\langle E_0[t_0]\rangle] \breve{\mathcal{R}}^{\mathsf{NF}\searrow} F'_1[t'_1\{E_1/k\}]$, hence the result holds.

Case: $F_0[t_0] \breve{\mathcal{R}} F_1[t_1]$ *with* $F_0[x] \breve{\mathcal{R}}^{\searrow} F_1[x]$ *(x fresh),* $t_0 \breve{\mathcal{R}} t_1$*, and* t_0*,* t_1 *delimited.* The term $F_0[t_0]$ is a normal form if t_0 is an open stuck term. We assume $F_0[x]$ and $F_1[x]$ are both delimited, the case where they are not is treated similarly (by replacing F_0, F_1 with $\langle k\, F_0\rangle$, $\langle k\, F_1\rangle$ for a fresh k). The terms t_0 and t_1 are both delimited, so by the induction hypothesis, there exists an open term t'_1 such that $t_1 \Downarrow_{\text{v}} t'_1$ and $t_0 \breve{\mathcal{R}}^{\mathsf{NF}\searrow} t'_1$. Consequently, we have $F_1[t_1] \Downarrow_{\text{v}} F_1[t'_1]$ with $F_0[t_0] \breve{\mathcal{R}}^{\mathsf{NF}\searrow} F_1[t'_1]$ by Lemma A.36, hence the result holds.

We now suppose $m > 0$. The cases $t_0 \mathrel{\mathcal{R}} t_1$ and $t_0 = t_1$ are still straightforward.

Case: $\mathcal{S}k.t_0 \breve{\mathcal{R}} \mathcal{S}k.t_1$ *with* $t_0 \breve{\mathcal{R}} t_1$*.* Let k' be a fresh variable. Then $\langle k'\, \mathcal{S}k.t_0\rangle \rightarrow_{\text{v}} \langle t_0\{\lambda x.\langle k'\, x\rangle/k\}\rangle \Downarrow_{\text{v}} t'_0$, and $\langle k'\, \mathcal{S}k.t_1\rangle \rightarrow_{\text{v}} \langle t_1\{\lambda x.\langle k'\, x\rangle/k\}\rangle$. The two terms $\langle t_0\{\lambda x.\langle k'\, x\rangle/k\}\rangle$ and $\langle t_1\{\lambda x.\langle k'\, x\rangle/k\}\rangle$ are related by $\breve{\mathcal{R}}$, and the former evaluates to a normal form in $m-1$ steps or less, therefore we can conclude by using the induction hypothesis.

Case: $t_0\{v_0/x\} \breve{\mathcal{R}} t_1\{v_1/x\}$ *with* $t_0 \breve{\mathcal{R}} t_1$ *and* $v_0 \breve{\mathcal{R}}^{\mathsf{NF}\searrow} v_1$*.* The reasoning is the same as in the corresponding case of the proof of Lemma A.21. If t_0 and t_1 are not both delimited, the reasoning is applied to $\langle k\, t_0\rangle$, $\langle k\, t_1\rangle$ (k fresh) instead of t_0, t_1.

Case: $t_0\{E_0/k\} \breve{\mathcal{R}} t_1\{E_1/k\}$ *with* $t_0 \breve{\mathcal{R}} t_1$ *and* $\langle E_0[x]\rangle \breve{\mathcal{R}} \langle E_1[x]\rangle$ *(x fresh).* The reasoning is the same as in the corresponding case of the proof of Lemma A.23 (again, eventually applied to $\langle k\, t_0\rangle$, $\langle k\, t_1\rangle$ with k fresh instead of t_0, t_1).

Case: $F_0[t_0] \breve{\mathcal{R}} F_1[t_1]$ *with* $F_0[x] \breve{\mathcal{R}}^{\searrow} F_1[x]$ *(x fresh),* $t_0 \breve{\mathcal{R}} t_1$*, and* t_0*,* t_1 *delimited.* We assume $F_0[x]$ and $F_1[x]$ are both delimited, the proof is the same if they are not (by replacing F_0 and F_1 with $\langle k\, F_0\rangle$ and $\langle k\, F_1\rangle$).

Because $F_0[t_0]$ evaluates to a normal form t'_0, we also have $t_0 \Downarrow_{\text{v}} t''_0$. If t''_0 is an open stuck term, then by the induction hypothesis, there exists an open stuck term t_1'' such that $t_1 \Downarrow_{\mathrm{v}} t_1''$ and $t_0'' \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} t_1''$. We have $F_1[t_1] \Downarrow_{\mathrm{v}} F_1[t_1'']$, and because t_0 and t_1 are delimited, t_0'' and t_1'' are also delimited. We can apply Lemma A.36, which gives us $F_0[t_0''] \mathrel{\breve{\mathcal{R}}^{\mathsf{NF}\searrow}} F_1[t_1'']$, as wished.

Finally, if t_0'' is a value v_0, then by the induction hypothesis, there exists v_1 such that $t_1 \Downarrow_{\mathrm{v}} v_1$ and $v_0 \mathrel{\widehat{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1$. Then we have $F_0[t_0] \rightarrow_{\mathrm{v}}^* F_0[v_0] = F_0[x]\{v_0/x\}$ and $F_1[t_1] \rightarrow_{\mathrm{v}}^* F_0[x]\{v_0/x\}$, and from there we conclude as in the substitutivity case.

Case: $E_0[t_0] \mathrel{\breve{\mathcal{R}}} E_1[t_1]$ *with* $E_0[x] \mathrel{\breve{\mathcal{R}}} E_1[x]$ *(x fresh) and* $t_0 \mathrel{\breve{\mathcal{R}}} t_1$. Let k be a fresh variable. If t_0 and t_1 are both delimited, then t_0 evaluates to a value or an open stuck term. We can then conclude the same way as in the corresponding case of the proof of Lemma A.21.

Suppose t_0 and t_1 are not both delimited, and let k' be a fresh variable. Because $\langle k\, E_0[t_0]\rangle$ evaluates to a normal form t_0', $\langle k'\, t_0\rangle$ also evaluates to a normal form t_0''. By the induction hypothesis, there exists t_1'' such that $\langle k'\, t_0\rangle \Downarrow_{\mathrm{v}} t_1''$ and $t_0'' \mathrel{\breve{\mathcal{R}}^{\mathsf{NF}\searrow}} t_1''$. If t_0'' is not of the form $F_0[\langle k'\, v_0\rangle]$, then we have $\langle k\, E_0[t_0]\rangle \Downarrow_{\mathrm{v}} t_0''\{k\ E_0/k'\}$, $\langle k\, E_1[t_1]\rangle \Downarrow_{\mathrm{v}} t_1''\{k\ E_1/k'\}$, with $t_0''\{k\ E_0/k'\} \mathrel{\breve{\mathcal{R}}^{\mathsf{NF}\searrow}} \langle k\, E_0[t_0]\rangle \Downarrow_{\mathrm{v}} t_1''\{k\ E_1/k'\}$, hence the result holds.

If $t_0'' = F_0[\langle k'\, v_0\rangle]$, then $t_1'' = F_1[\langle k'\, v_1\rangle]$ with $v_0 \mathrel{\breve{\mathcal{R}}^{\mathsf{NF}\searrow}} v_1$ and $F_0[\langle\Box\rangle] \mathrel{\breve{\mathcal{R}}^{\mathsf{C}\searrow}} F_1[\langle\Box\rangle]$. We therefore have $\langle k\, E_0[t_0]\rangle \rightarrow_{\mathrm{v}}^* F_0\{k\ E_0/k'\}[\langle E_0[v_0\{k\ E_0/k'\}]\rangle] \Downarrow_{\mathrm{v}} t_0'$, and $\langle k\, E_1[t_1]\rangle \rightarrow_{\mathrm{v}}^* F_1\{k\ \ E_1/k'\}[\langle E_1[v_1\{k\ E_1/k'\}]\rangle]$. If $\langle k\, E_0[t_0]\rangle$ reduces to $F_0\{k\ E_0/k'\}[\langle E_0[v_0\{k\ E_0/k'\}]\rangle]$ in at least one step, then we can conclude using the induction hypothesis, because we have $F_0\{k\ E_0/k'\}[\langle E_0[v_0\{k\ E_0/k'\}]\rangle] \mathrel{\breve{\mathcal{R}}} F_1\{k\ E_1/k'\}[\langle E_1[v_1\{k\ E_1/k'\}]\rangle]$. Otherwise, $t_0 = v_0$, and we have $\langle k\, E_0[v_0]\rangle \Downarrow_{\mathrm{v}} t_0'$, i.e., $\langle k\, E_0[x]\rangle\{v_0/x\} \Downarrow_{\mathrm{v}} t_0'$; we can conclude as in the substituvity case, because we have $E_0[x] \mathrel{\breve{\mathcal{R}}} E_1[x]$.

Case: $F_0[\langle E_0[t_0]\rangle] \mathrel{\breve{\mathcal{R}}} F_1[\langle E_1[t_1]\rangle]$ *with* $F_0[x] \mathrel{\breve{\mathcal{R}}} F_1[x]$, $\langle E_0[x]\rangle \mathrel{\breve{\mathcal{R}}} \langle E_1[x]\rangle$ *(x fresh), and* $t_0 \mathrel{\breve{\mathcal{R}}} t_1$. If t_0 and t_1 are both delimited, then the proof is the same as in the corresponding case of the proof of Lemma A.21. Otherwise, the proof is the same as in the previous case (with an extra layer of context).

B. Applicative Bisimilarity

Lemma B.1. *If $t_0 \mathbb{A}^\bullet t_1$, then there exists a substitution σ which closes t_0 and t_1 such that $t_0\sigma\ (\mathbb{A}^\bullet)^c\ t_1\sigma$, and the size of the derivation of $t_0\sigma\ (\mathbb{A}^\bullet)^c\ t_1\sigma$ is equal to the size of the derivation of $t_0 \mathbb{A}^\bullet t_1$.*

Proof. By induction on $t_0 \mathbb{A}^\bullet t_1$. Suppose we have $t_0 \mathbb{A}^\circ t_1$. Let σ be a substitution which closes t_0 and t_1; we have $t_0\sigma \mathbb{A}^\circ t_1\sigma$. The remaining cases are easy using induction.

Lemma B.2. *Let t be a closed term. If $t \xrightarrow{\alpha} t'$, then for all $\alpha\ (\mathbb{A}^\bullet)^c\ \alpha'$, there exists t'' such that $t \xrightarrow{\alpha'} t''$ and $t'\ (\mathbb{A}^\bullet)^c\ t''$.*

PROOF. Easy by induction on $t \xrightarrow{\alpha} t'$.

Lemma B.3. *If* $\lambda x.t_0 \ (\mathbb{A}^\bullet)^c \ \lambda x.t_1$ *and* $v_0 \ (\mathbb{A}^\bullet)^c \ v_1$ *then* $t_0\{v_0/x\} \ (\mathbb{A}^\bullet)^c \ t_1\{v_1/x\}$.

PROOF. By induction on $\lambda x.t_0 \ (\mathbb{A}^\bullet)^c \ \lambda x.t_1$.

Suppose $\lambda x.t_0 \ \mathbb{A} \ \lambda x.t_1$. We have $\lambda x.t_0 \xrightarrow{v_0} t_0\{v_0/x\}$, so by Lemma B.2, there exists t'_0 such that $\lambda x.t_0 \xrightarrow{v_1} t'_0$ and $t_0\{v_0/x\} \ \mathbb{A}^\bullet \ t'_0$. By bisimilarity, there exists t'_1 such that $\lambda x.t_1 \xrightarrow{v_1} t'_1$ and $t'_0 \ \mathbb{A} \ t'_1$. The only possible outcome is $t'_1 = t_1\{v_1/x\}$, therefore we have $t_0\{v_0/x\} \ \mathbb{A}^\bullet\mathbb{A} \ t_1\{v_1/x\}$. Because the considered terms are closed, we have the required result.

Suppose $\lambda x.t_0 \ \mathbb{A}^\bullet\mathbb{A} \ \lambda x.t_1$. The result follows from the induction hypothesis and a similar reasoning as in the first case.

Suppose $\lambda x.t_0 \ \widehat{\mathbb{A}^\bullet} \ \lambda x.t_1$, which comes from $t_0 \ \mathbb{A}^\bullet \ t_1$. We have $t_0\{v_0/x\} \ \mathbb{A}^\bullet \ t_1\{v_1/x\}$ by substitutivity of $\mathbb{A}^\bullet$.

Lemma B.4. *If* $v_0 \ (\mathbb{A}^\bullet)^c \ t_1$*, then there exists* v_1 *such that* $t_1 \stackrel{\tau}{\Rightarrow} v_1$ *and* $v_0 \ (\mathbb{A}^\bullet)^c \ v_1$.

PROOF. By induction on $v_0 \ (\mathbb{A}^\bullet)^c \ t_1$. Suppose $v_0 \ \mathbb{A} \ t_1$. Let $v_0 = \lambda x.t_0$; for all v, we have $v_0 \xrightarrow{v} t_0\{v/x\}$. By bisimilarity, there exists t_1^v such that $v_0 \stackrel{v}{\Rightarrow} t_1^v$ and $t_0\{v/x\} \ \mathbb{A} \ t_1^v$. Therefore there exists $v_1 = \lambda x.t'_1$ such that $t_1 \stackrel{\tau}{\Rightarrow} v_1 \xrightarrow{v} t'_1\{v/x\}$ and $t_0\{v/x\} \ \mathbb{A} \ t'_1\{v/x\}$. Because this holds for all v, we have $t_0 \ \mathbb{A}^\circ \ t'_1$, therefore we have $t_0 \ \mathbb{A}^\bullet \ t'_1$. From this observation, we deduce $\lambda x.t_0 \ (\mathbb{A}^\bullet)^c \ \lambda x.t'_1$, as wished. The case $v_0 \ \mathbb{A}^\bullet\mathbb{A} \ t_1$ relies on induction and a similar reasoning as in the first case. If $v_0 \ \widehat{\mathbb{A}^\bullet} \ t_1$, then t_1 is a value and we can conclude directly.

Lemma B.5. *If* $E_0 \ \mathbb{A}^\bullet \ E_1$ *and* $t_0 \ \mathbb{A}^\bullet \ t_1$ *then* $E_0[t_0] \ \mathbb{A}^\bullet \ E_1[t_1]$.

PROOF. Easy by induction on $E_0 \ \mathbb{A}^\bullet \ E_1$.

Lemma B.6 (Lemma 5.12 in the article). *If* $t_0 \ (\mathbb{A}^\bullet)^c \ t_1$ *and* $t_0 \xrightarrow{\alpha} t'_0$*, then for all* $\alpha \ (\mathbb{A}^\bullet)^c \ \alpha'$*, there exists* t'_1 *such that* $t_1 \stackrel{\alpha'}{\Rightarrow} t'_1$ *and* $t'_0 \ (\mathbb{A}^\bullet)^c \ t'_1$.

PROOF. By induction on the size of the derivation of $t_0 \ (\mathbb{A}^\bullet)^c \ t_1$.

If $t_0 \ \mathbb{A}^\circ \ t_1$, then we have $t_0 \ \mathbb{A} \ t_1$ because we work with closed terms. By Lemma B.2, there exists t''_0 such that $t_0 \xrightarrow{\alpha'} t''_0$ and $t'_0 \ (\mathbb{A}^\bullet)^c \ t''_0$. By bisimilarity, there exists t'_1 such that $t_1 \stackrel{\alpha'}{\Rightarrow} t'_1$ and $t''_0 \ \mathbb{A} \ t'_1$ (i.e., $t''_0 \ \mathbb{A}^\circ \ t'_1$ because the terms are closed). Therefore we have $t'_0 \ (\mathbb{A}^\bullet)^c\mathbb{A}^\circ \ t'_1$, i.e., $t'_0 \ (\mathbb{A}^\bullet)^c \ t'_1$, as required.

If $t_0 \ \mathbb{A}^\bullet \ t_2 \ \mathbb{A}^\circ \ t_1$, then by Lemma B.1, there exists σ such that $t_0\sigma \ (\mathbb{A}^\bullet)^c \ t_2\sigma$ and the size of the derivation of $t_0\sigma \ (\mathbb{A}^\bullet)^c \ t_2\sigma$ is the same as for $t_0 \ (\mathbb{A}^\bullet)^c \ t_2$. Because t_0 and t_1 are closed, and by definition of $\mathbb{A}^\circ$, we have in fact $t_0 \ (\mathbb{A}^\bullet)^c \ t_2\sigma \ \mathbb{A} \ t_1$. By induction hypothesis, there exists t'_2 such that $t_2\sigma \stackrel{\alpha'}{\Rightarrow} t'_2$ and $t'_0 \ (\mathbb{A}^\bullet)^c \ t'_2$. By bisimilarity, there exists t'_1 such that $t_1 \stackrel{\alpha'}{\Rightarrow} t'_1$ and $t'_2 \ \mathbb{A} \ t'_1$ (i.e., $t'_2 \ \mathbb{A}^\circ \ t'_1$ because the terms are closed). Therefore we have $t'_0 \ (\mathbb{A}^\bullet)^c\mathbb{A}^\circ \ t'_1$, i.e.,

$t'_0 \ (\mathbb{A}^\bullet)^c \ t'_1$, as required.

If $t_0 \ \widehat{\mathbb{A}^\bullet} \ t_1$, then we distinguish several cases, depending on the outermost operator.

Suppose $t_0 = \lambda x.s_0$ and $t_1 = \lambda x.s_1$ with $s_0 \ \mathbb{A}^\bullet \ s_1$. The only possible transition is $t_0 \xrightarrow{v} s_0\{v/x\}$. We have $t_1 \xrightarrow{v'} s_1\{v'/x\}$. By substitutivity of $\mathbb{A}^\bullet$, we have $s_0\{v/x\} \ \mathbb{A}^\bullet \ s_1\{v'/x\}$, and because x is the only free variable of s_0 and s_1, we have $s_0\{v/x\} \ (\mathbb{A}^\bullet)^c \ s_1\{v'/x\}$, as required.

Suppose $t_0 = \mathcal{S}k.s_0$ and $t_1 = \mathcal{S}k.s_1$ with $s_0 \ \mathbb{A}^\bullet \ s_1$. The only possible transition is $t_0 \xrightarrow{E} \langle s_0\{\lambda x.\langle E[x]\rangle/k\}\rangle$. We have $t_1 \xrightarrow{E'} \langle s_1\{\lambda x.\langle E'[x]\rangle/k\}\rangle$. Because $\mathbb{A}^\bullet$ is compatible and by Lemma B.5, we have $\lambda x.\langle E[x]\rangle \ (\mathbb{A}^\bullet)^c \ \lambda x.\langle E'[x]\rangle$. Therefore, by substitutivity of $\mathbb{A}^\bullet$, we have $\langle s_0\{\lambda x.\langle E[x]\rangle/k\}\rangle \ \mathbb{A}^\bullet \ s_1\{\lambda x.\langle E'[x]\rangle/k\}$, and because k is the only free variable of s_0 and s_1, we have $\langle s_0\{\lambda x.\langle E[x]\rangle/k\}\rangle \ (\mathbb{A}^\bullet)^c$ $\langle s_1\{\lambda x.\langle E'[x]\rangle/k\}\rangle$, as required.

Suppose $t_0 = (\lambda x.s_0) \, v_0$ and $t_1 = t_1^1 \, t_1^2$ with $\lambda x.s_0 \ (\mathbb{A}^\bullet)^c \ t_1^1$ and $v_0 \ (\mathbb{A}^\bullet)^c \ t_1^2$. The only possible transition is $t_0 \xrightarrow{\tau} s_0\{v_0/x\}$. By Lemma B.4, there exists $\lambda x.s_1$, v_1 such that $t_1^1 \stackrel{\tau}{\Rightarrow} \lambda x.s_1$, $t_1^2 \stackrel{\tau}{\Rightarrow} v_1$, $\lambda x.s_0 \ (\mathbb{A}^\bullet)^c \ \lambda x.s_1$, and $v_0 \ (\mathbb{A}^\bullet)^c$ v_1. From $t_1^1 \stackrel{\tau}{\Rightarrow} \lambda x.s_1$ and $t_1^2 \stackrel{\tau}{\Rightarrow} v_1$, we can deduce $t_1 \stackrel{\tau}{\Rightarrow} s_1\{v_1/x\}$, and from $\lambda x.s_0 \ (\mathbb{A}^\bullet)^c \ \lambda x.s_1$ and $v_0 \ (\mathbb{A}^\bullet)^c \ v_1$, we have $s_0\{v_0/x\} \ \mathbb{A}^\bullet \ s_1\{v_1/x\}$ by Lemma B.3. Hence, we have the required result.

Suppose $t_0 = v_0 \, s_0$ and $t_1 = t_1^1 \, s_1$ with $v_0 \ (\mathbb{A}^\bullet)^c \ t_1^1$ and $s_0 \ (\mathbb{A}^\bullet)^c \ s_1$. By Lemma B.4, there exists v_1 such that $t_1^1 \stackrel{\tau}{\Rightarrow} v_1$ and $v_0 \ (\mathbb{A}^\bullet)^c \ v_1$. We have two possible cases; in the first one, we have $t_0 \xrightarrow{\tau} v_0 s'_0$, where $s_0 \xrightarrow{\tau} s'_0$. By the induction hypothesis, there exists s'_1 such that $s_1 \stackrel{\tau}{\Rightarrow} s'_1$ and $s'_0 \ (\mathbb{A}^\bullet)^c \ s'_1$. Therefore we have $t_1 \stackrel{\tau}{\Rightarrow} v_1 s'_1$, and because $(\mathbb{A}^\bullet)^c$ is compatible, we have $v_0 s'_0 \ (\mathbb{A}^\bullet)^c \ v_1 s'_1$, hence the result holds. In the second case, we have $t_0 \xrightarrow{E} s'_0$, where $s_0 \xrightarrow{E[v_0 \ \Box]} s'_0$. Let E' such that $E \ (\mathbb{A}^\bullet)^c \ E'$; then $E[v_0 \ \Box] \ (\mathbb{A}^\bullet)^c \ E'[v_1 \ \Box]$ holds. Therefore, by the induction hypothesis, there exists s'_1 such that $s_1 \xRightarrow{E'[v_1 \ \Box]} s'_1$ and $s'_0 \ (\mathbb{A}^\bullet)^c \ s'_1$. The transition $s_1 \xRightarrow{E'[v_1 \ \Box]} s'_1$ implies $v_1 \, s_1 \xRightarrow{E'} s'_1$, hence we have $t_1 \stackrel{\tau}{\Rightarrow} v_1 \, s_1 \xRightarrow{E'} s'_1$, i.e., $t_1 \xRightarrow{E'} s'_1$, Because $s'_0 \ (\mathbb{A}^\bullet)^c \ s'_1$ holds, we have the required result.

Suppose $t_0 = t_0^1 \, t_0^2$ and $t_1 = t_1^1 \, t_1^2$ with $t_0^1 \ (\mathbb{A}^\bullet)^c \ t_1^1$ and $t_0^2 \ (\mathbb{A}^\bullet)^c \ t_1^2$. We have two possible cases. First, suppose $t_0 \xrightarrow{E} t'_0$, where $t_0^1 \xrightarrow{E[\Box \ t_0^2]} t'_0$. Let E' such that $E \ (\mathbb{A}^\bullet)^c \ E'$; then $E[\Box \ t_0^2] \ (\mathbb{A}^\bullet)^c \ E'[\Box \ t_1^2]$ holds. Therefore, by the induction hypothesis, there exists t'_1 such that $t_1^1 \xRightarrow{E'[\Box \ t_1^2]} t'_1$ and $t'_0 \ (\mathbb{A}^\bullet)^c \ t'_1$. The transition $t_1^1 \xRightarrow{E'[\Box \ t_1^2]} t'_1$ implies $t_1^1 \, t_1^2 \xRightarrow{E'} t'_1$, hence we have the required result. Next, suppose $t_0 \xrightarrow{\tau} {t_0^1}' \, t_0^2$, where $t_0^1 \xrightarrow{\tau} {t_0^1}'$. By the induction hypothesis, there exists ${t_1^1}'$ such that $t_1^1 \stackrel{\tau}{\Rightarrow} {t_1^1}'$ and ${t_0^1}' \ (\mathbb{A}^\bullet)^c \ {t_1^1}'$. Therefore we have $t_1 \stackrel{\tau}{\Rightarrow} {t_1^1}' \, t_1^2$, and because $(\mathbb{A}^\bullet)^c$ is compatible, we have ${t_0^1}' \, t_0^2 \ (\mathbb{A}^\bullet)^c \ {t_1^1}' \, t_1^2$, hence the result holds.

Suppose $t_0 = \langle v_0\rangle$ and $t_1 = \langle s_1\rangle$ with $v_0 \ (\mathbb{A}^\bullet)^c \ s_1$. The only possible transition is $t_0 \xrightarrow{\tau} v_0$. By Lemma B.4, there exists v_1 such that $s_1 \stackrel{\tau}{\Rightarrow} v_1$ and

$v_0 \ (\mathbb{A}^\bullet)^c \ v_1$. We have $t_1 \stackrel{\tau}{\Rightarrow} v_1$, with $v_0 \ (\mathbb{A}^\bullet)^c \ v_1$, hence the result holds. The case $t_0 = \langle s_0 \rangle$, where s_0 is not a value, is easy using the induction hypothesis and compatibility of $(\mathbb{A}^\bullet)^c$.

Lemma B.7. *The relation $((\mathbb{A}^\bullet)^c)^*$ is a bisimulation.*

Proof. We know that $((\mathbb{A}^\bullet)^c)^*$ is symmetric by Lemma 5.13, so it is enough to prove that $((\mathbb{A}^\bullet)^c)^*$ is a simulation. Let $t_0((\mathbb{A}^\bullet)^c)^* t_1$; there exists an integer k such that $t_0 \ ((\mathbb{A}^\bullet)^c)^k \ t_1$. Let $(t_0^i)_{i \in \{1 \dots k\}}$ be the terms such that $t_0 \ (\mathbb{A}^\bullet)^c \ t_0^1 \ (\mathbb{A}^\bullet)^c \ t_0^2 \dots t_0^{k-1} \ (\mathbb{A}^\bullet)^c \ t_0^k = t_1$. Let $t_0 \xrightarrow{\alpha} t_0'$. We prove by induction on $i \in 1 \dots k$ that there exists ${t_0^i}'$ such that $t_0^i \stackrel{\alpha}{\Rightarrow} {t_0^i}'$ and $t_0' \ ((\mathbb{A}^\bullet)^c)^i \ {t_0^i}'$. Suppose $i = 1$. We have $\alpha \ (\mathbb{A}^\bullet)^c \ \alpha$ so by Lemma B.6, there exists ${t_0^1}'$ such that $t_0^1 \stackrel{\alpha}{\Rightarrow} {t_0^1}'$ and $t_0' \ (\mathbb{A}^\bullet)^c \ {t_0^1}'$, as wished. The case $1 < i \leq k$ is easy by induction. In particular, for $i = k$, we have $t_1 = t_0^k \stackrel{\alpha}{\Rightarrow} {t_0^k}'$ and $t_0'((\mathbb{A}^\bullet)^c)^k {t_0^k}'$. We have the required result because $((\mathbb{A}^\bullet)^c)^k \subseteq ((\mathbb{A}^\bullet)^c)^*$.

C. Environmental Bisimilarity

C.1. Soundness Proof for the Relaxed Semantics

Lemma C.1. *Let $\mathcal{R}$ be a relation on closed terms. If $t_0 \ \dot{\mathcal{R}} \ t_1$ (where t_0 and t_1 are potentially open terms) and $v_0 \ \dot{\mathcal{R}}^{\mathrm{v}} \ v_1$, then $t_0\{v_0/x\} \ \dot{\mathcal{R}} \ t_1\{v_1/x\}$.*

Proof. By induction on $t_0 \ \dot{\mathcal{R}} \ t_1$.

Lemma C.2. *Let $\mathcal{E}$ be an environment. Suppose $t_0 \ \dot{\mathcal{E}} \ t_1$. If t_0 is a value, then so is t_1.*

Proof. Easy by induction on $t_0 \ \dot{\mathcal{E}} \ t_1$.

Lemma C.3. *Let $\mathcal{E}$ be an environment*

- *If $v_0 \approx_{\mathcal{E}} v_1$, then $C[v_0] \approx_{\dot{\mathcal{E}}^{\mathrm{v}}} C[v_1]$.*
- *If $t_0 \approx_{\mathcal{E}} t_1$, then $F[t_0] \approx_{\dot{\mathcal{E}}^{\mathrm{v}}} F[t_1]$.*

Let $\mathcal{Y}$ be an environmental bisimulation. We define

$$\begin{aligned}
\mathcal{X} &= \mathcal{X}_1 \cup \mathcal{X}_2 \cup \{\dot{\mathcal{E}}^{\mathrm{v}} \mid \mathcal{E} \in \mathcal{Y}\} \\
\mathcal{X}_1 &= \{(\dot{\mathcal{E}}^{\mathrm{v}}, F_0[t_0], F_1[t_1]) \mid t_0 \ \mathcal{Y}_{\mathcal{E}} \ t_1, F_0 \ \ddot{\mathcal{E}} \ F_1\} \\
\mathcal{X}_2 &= \{(\dot{\mathcal{E}}^{\mathrm{v}}, t_0, t_1) \mid \mathcal{E} \in \mathcal{Y}, t_0 \ \dot{\mathcal{E}} \ t_1\}
\end{aligned}$$

In $\mathcal{X}_2$, we build the closed terms (t_0, t_1) out of pairs of values. We first prove a preliminary lemma about $\mathcal{X}$.

Lemma C.4. *Let $\mathcal{E} \in \mathcal{Y}$. If $\lambda x.t_0 \ \dot{\mathcal{E}} \ \lambda x.t_1$ and $v_0 \ \dot{\mathcal{E}}^{\mathrm{v}} \ v_1$ then $t_0\{v_0/x\} \ \mathcal{X}_{\dot{\mathcal{E}}^{\mathrm{v}}} \ t_1\{v_1/x\}$.*

PROOF. We have two cases. If $\lambda x.t_0 \ \mathcal{E} \ \lambda x.t_1$, then $t_0\{v_0/x\} \ \mathcal{Y}_{\mathcal{E}} \ t_1\{v_1/x\}$ (because $\mathcal{Y}$ is a bisimulation), which implies $t_0\{v_0/x\} \ \mathcal{X}_{\dot{\mathcal{E}}^{\mathrm{v}}} \ t_1\{v_1/x\}$ (more precisely, the terms are in $\mathcal{X}_1$). If $t_0 \ \dot{\mathcal{E}} \ t_1$ with $\mathsf{fv}(t_0) \cup \mathsf{fv}(t_1) \subseteq \{x\}$, then we have $t_0\{v_0/x\} \ \dot{\mathcal{E}} \ t_1\{v_1/x\}$ by Lemma C.1, which implies $t_0\{v_0/x\} \ \mathcal{X}_{\dot{\mathcal{E}}^{\mathrm{v}}} \ t_1\{v_1/x\}$ (more precisely, the terms are in $\mathcal{X}_2$).

We now prove Lemma C.3 by showing that $\mathcal{X}$ is a bisimulation up to environment.

PROOF. We first prove the bisimulation for the elements in $\mathcal{X}_2$ (for these, we do not need the "up to environment"). Let $t_0 \ \dot{\mathcal{E}} \ t_1$, with $\mathcal{E} \in \mathcal{Y}$. If t_0 is a value, then t_1 is a value (by Lemma C.2), and we have $\dot{\mathcal{E}}^{\mathrm{v}} \cup \{(t_0, t_1)\} = \dot{\mathcal{E}}^{\mathrm{v}} \in \mathcal{X}$. We now consider the case where t_0 is not a value. We proceed by induction on $t_0 \ \dot{\mathcal{E}} \ t_1$.

Suppose $t_0 = t_0^1 \, t_0^2$ and $t_1 = t_1^1 \, t_1^2$ with $t_0^1 \ \dot{\mathcal{E}} \ t_1^1$ and $t_0^2 \ \dot{\mathcal{E}} \ t_1^2$. We have several cases to consider.

- Assume $t_0^1 \rightarrow_{\mathrm{v}} t_0^{1\prime}$, so that $t_0 \rightarrow_{\mathrm{v}} t_0^{1\prime} \, t_0^2$. By the induction hypothesis, there exists $t_1^{1\prime}$ such that $t_1^1 \rightarrow_{\mathrm{v}}^* t_1^{1\prime}$ and $t_0^{1\prime} \ \mathcal{X}_{\dot{\mathcal{E}}^{\mathrm{v}}} \ t_1^{1\prime}$. From $t_0^2 \ \dot{\mathcal{E}} \ t_1^2$ and $t_0^{1\prime} \ \mathcal{X}_{\dot{\mathcal{E}}^{\mathrm{v}}} \ t_1^{1\prime}$, we can deduce $t_0^{1\prime} \, t_0^2 \ \mathcal{X}_{\dot{\mathcal{E}}^{\mathrm{v}}} \ t_1^{1\prime} \, t_1^2$ by definition of $\mathcal{X}$. We also have $t_1 \rightarrow_{\mathrm{v}}^* t_1^{1\prime} \, t_1^2$, hence the result holds.

- Assume $t_0^1 = v_0$ and $t_0^2 \rightarrow_{\mathrm{v}} t_0^{2\prime}$, so that $t_0 \rightarrow_{\mathrm{v}} v_0 \, t_0^{2\prime}$. Then t_1^1 is also a value v_1 according to Lemma C.2. By the induction hypothesis, there exists $t_1^{2\prime}$ such that $t_1^2 \rightarrow_{\mathrm{v}}^* t_1^{2\prime}$ and $t_0^{2\prime} \ \mathcal{X}_{\dot{\mathcal{E}}^{\mathrm{v}}} \ t_1^{2\prime}$. From $v_0 \ \dot{\mathcal{E}} \ v_1^2$ and $t_0^{2\prime} \ \mathcal{X}_{\dot{\mathcal{E}}^{\mathrm{v}}} \ t_1^{2\prime}$, we can deduce $v_0 \, t_0^{2\prime} \ \mathcal{X}_{\dot{\mathcal{E}}^{\mathrm{v}}} \ v_1 \, t_1^{2\prime}$ by definition of $\mathcal{X}$. We also have $t_1 \rightarrow_{\mathrm{v}}^* v_1 \, t_1^{2\prime}$, hence the result holds.

- Assume $t_0^1 = \lambda x.t_0'$ and $t_0^2 = v_0$, so that $t_0 \rightarrow_{\mathrm{v}} t_0'\{v_0/x\}$. By Lemma C.2, t_1^1 is a value $\lambda x.t_1'$ and t_1^2 is a value v_1. We have $t_1 \rightarrow_{\mathrm{v}} t_0'\{v_0/x\}$, and by Lemma C.4, we have $t_0'\{v_0/x\} \ \mathcal{X}_{\dot{\mathcal{E}}^{\mathrm{v}}} \ t_1'\{v_1/x\}$, hence the result holds.

- Assume $t_0^1 = v_0$ and $t_0^2 = E_0[\mathcal{S}k.t_0']$. By Lemma C.2, t_1^1 is a value v_1, and by the induction hypothesis, there exists $E_1[\mathcal{S}k.t_1']$ such that $t_1^2 \Downarrow_{\mathrm{v}} E_1[\mathcal{S}k.t_1']$ and for all $E_0' \ (\ddot{\dot{\mathcal{E}}}^{\mathrm{v}}) \ E_1'$, we have $\langle t_0'\{\lambda x.\langle E_0'[E_0[x]]\rangle/k\}\rangle \ \mathcal{X}_{\dot{\mathcal{E}}^{\mathrm{v}}} \ \langle t_1'\{\lambda x.\langle E_1'[E_1[x]]\rangle/k\}\rangle$. In particular, $E_0' \ (\ddot{\dot{\mathcal{E}}}^{\mathrm{v}}) \ E_1'$ and $v_0 \ \dot{\mathcal{E}} \ v_1$ implies $E_0'[v_0 \ \Box] \ (\ddot{\dot{\mathcal{E}}}^{\mathrm{v}}) \ E_1'[v_1 \ \Box]$, therefore we have $\langle t_0'\{\lambda x.\langle E_0'[v_0 \ E_0[x]]\rangle/k\}\rangle \ \mathcal{X}_{\dot{\mathcal{E}}^{\mathrm{v}}} \ \langle t_1'\{\lambda x.\langle E_1'[v_1 \ E_1[x]]\rangle/k\}\rangle$, as wished. The reasoning is similar if $t_0^1 = E_0[\mathcal{S}k.t_0']$.

Suppose $t_0 = \langle t_0' \rangle$, $t_1 = \langle t_1' \rangle$ with $t_0' \ \dot{\mathcal{E}} \ t_1'$. We have two possibilities.

- Assume $t_0' \rightarrow_{\mathrm{v}} t_0''$, so that $t_0 \rightarrow_{\mathrm{v}} \langle t_0'' \rangle$. By the induction hypothesis, there exists t_1'' such that $t_1' \rightarrow_{\mathrm{v}}^* t_1''$ and $t_0'' \ \mathcal{X}_{\dot{\mathcal{E}}^{\mathrm{v}}} \ t_1''$. By definition of $\mathcal{X}$, we have $\langle t_0'' \rangle \ \mathcal{X}_{\dot{\mathcal{E}}^{\mathrm{v}}} \ \langle t_1'' \rangle$, and furthermore $t_0 \rightarrow_{\mathrm{v}}^* \langle t_1'' \rangle$, we therefore have the required result.

- Assume $t'_0 = E_0[\mathcal{S}k.t''_0]$, so that $t_0 \rightarrow_{\mathrm{v}} \langle t''_0\{\lambda x.\langle E_0[x]\rangle/k\}\rangle$. By the induction hypothesis, there exists $E_1[\mathcal{S}k.t''_1]$ such that $t'_1 \Downarrow_{\mathrm{v}} E_1[\mathcal{S}k.t''_1]$ and $\langle t''_0\{\lambda x.\langle E_0[x]\rangle/k\}\rangle \; \mathcal{X}_{\dot{\mathcal{E}}^{\mathrm{v}}} \; \langle t''_1\{\lambda x.\langle E_1[x]\rangle/k\}\rangle$. Because we also have $t_1 \rightarrow_{\mathrm{v}} \langle t''_1\{\lambda x.\langle E_1[x]\rangle/k\}\rangle$, the result holds.

Suppose $t_0 = \mathcal{S}k.t'_0$ with $t_1 = \mathcal{S}k.t'_1$ and $t'_0 \; \dot{\mathcal{E}} \; t'_1$. Let $E_0 \; (\ddot{\dot{\mathcal{E}}}^{\mathrm{v}}) \; E_1$. For a fresh x, we have $\lambda x.\langle E_0[x]\rangle \; \dot{\mathcal{E}} \; \lambda x.\langle E_1[x]\rangle$, which in turn implies $\langle t'_0\{\lambda x.\langle E_0[x]\rangle/k\}\rangle \; \mathcal{X}_{\dot{\mathcal{E}}^{\mathrm{v}}} \; \langle t'_1\{\lambda x.\langle E_1[x]\rangle/k\}\rangle$ by Lemma C.1 and compatibility of $\dot{\mathcal{E}}$. We therefore have the required result.

We now prove the bisimulation property (up to environment) for elements in $\mathcal{X}_1$. Let $F_0[t_0] \; \mathcal{X}_{\dot{\mathcal{E}}^{\mathrm{v}}} \; F_1[t_1]$, so that $t_0 \; \mathcal{Y}_{\mathcal{E}} \; t_1$ and $F_0 \; \ddot{\mathcal{E}} \; F_1$. If t_0 is a value v_0, then because $\mathcal{Y}$ is a bisimulation, there exists v_1 such that $t_1 \rightarrow^*_{\mathrm{v}} v_1$ and $\mathcal{E}' \overset{\mathsf{def}}{=} \mathcal{E} \cup \{(v_0, v_1)\} \in \mathcal{Y}$. We then have $F_1[t_1] \rightarrow^*_{\mathrm{v}} F_1[v_1]$, and the terms $F_0[v_0]$, $F_1[v_1]$ are similar to the one of $\mathcal{X}_2$. We can prove the bisimulation property with $F_0[v_0]$, $F_1[v_1]$ the same way we did with the terms in $\mathcal{X}_2$, except that we reason up to environment, because $\mathcal{E} \subseteq \mathcal{E}'$.

Suppose $t_0 \rightarrow_{\mathrm{v}} t'_0$, and so $F_0[t_0] \rightarrow_{\mathrm{v}} F_0[t'_0]$. Because $\mathcal{Y}$ is a bisimulation, there exists t'_1 such that $t_1 \rightarrow^*_{\mathrm{v}} t'_1$ and $t'_0 \; \mathcal{Y}_{\mathcal{E}} \; t'_1$. We therefore have $F_1[t_1] \rightarrow^*_{\mathrm{v}} F_1[t'_1]$ with $F_0[t'_0] \; \mathcal{X}_{\dot{\mathcal{E}}^{\mathrm{v}}} \; F_1[t'_1]$, as wished.

Suppose t_0 is a control stuck term $E_0[\mathcal{S}k.t'_0]$. We distinguish two cases. If F_0 is a pure context E'_0, then F_1 is also a pure context E'_1 (as one can easily check by induction on $F_0 \; \ddot{\mathcal{E}} \; F_1$). Let $E''_0 \; (\ddot{\dot{\mathcal{E}}}^{\mathrm{v}}) \; E''_1$; then we have $E''_0 \; \ddot{\mathcal{E}} \; E''_1$. Because $\mathcal{Y}$ is a bisimulation, there exists $E_1[\mathcal{S}k.t'_1]$ such that $t_1 \Downarrow_{\mathrm{v}} E_1[\mathcal{S}k.t'_1]$ and $\langle t'_0\{\lambda x.\langle E''_0[E'_0[E_0[x]]]\rangle/k\}\rangle \; \mathcal{Y}_{\mathcal{E}} \; \langle t'_1\{\lambda x.\langle E''_1[E'_1[E_1[x]]]\rangle/k\}\rangle$. Consequently, we have the required result. If F_0 is a context of the form $F'_0[\langle E'_0\rangle]$, then F_1 is of the form $F'_1[\langle E'_1\rangle]$ with $F'_0 \; \ddot{\mathcal{E}} \; F'_1$ and $E'_0 \; \ddot{\mathcal{E}} \; E'_1$ (again, by induction on $F_0 \; \ddot{\mathcal{E}} \; F_1$). Because $\mathcal{Y}$ is a bisimulation, there exists $E_1[\mathcal{S}k.t'_1]$ such that $t_1 \Downarrow_{\mathrm{v}} E_1[\mathcal{S}k.t'_1]$ and $\langle t'_0\{\lambda x.\langle E'_0[E_0[x]]\rangle/k\}\rangle \; \mathcal{Y}_{\mathcal{E}} \; \langle t'_1\{\lambda x.\langle E'_1[E_1[x]]\rangle/k\}\rangle$. But then $F_0[t_0] \rightarrow_{\mathrm{v}} F'_0[\langle t'_0\{\lambda x.\langle E'_0[E_0[x]]\rangle/k\}\rangle]$ and $F_1[t_1] \rightarrow_{\mathrm{v}} F'_1[\langle t'_1\{\lambda x.\langle E'_1[E_1[x]]\rangle/k\}\rangle]$, hence we have the required result.

We now prove the clause of the bisimulation about environments. The only environments in $\mathcal{X}$ are of the form $\dot{\mathcal{E}}^{\mathrm{v}}$. Let $\lambda x.t_0 \; \dot{\mathcal{E}}^{\mathrm{v}} \; \lambda x.t_1$ and $v_0 \; \dot{\mathcal{E}}^{\mathrm{v}} \; v_1$. By Lemma C.4, we have $t_0\{v_0/x\} \; \mathcal{X}_{\dot{\mathcal{E}}^{\mathrm{v}}} \; t_1\{v_1/x\}$, hence the result holds.

Lemma C.5. *If $\lambda x.t_0 \; \mathbb{E} \; \lambda x.t_1$, then $t_0\{v/x\} \; \mathbb{E} \; t_1\{v/x\}$.*

Proof. Let $\mathcal{E} \overset{\mathsf{def}}{=} \{(\lambda x.t_0, \lambda x.t_1)\}$. By bisimilarity, we have $\mathcal{E} \in \mathbb{E}$, which in turn implies $t_0\{v/x\} \approx_{\mathcal{E}} t_1\{v/x\}$ for all v. We then have $t_0\{v/x\} \; \mathbb{E} \; t_1\{v/x\}$ by weakening (Proposition 6.5).

Lemma C.6. *If $\lambda x.t_0 \; \dot{\mathbb{E}} \; t_1$, then there exists t'_1 such that $t_1 \Downarrow_{\mathrm{v}} \lambda x.t'_1$, $\lambda x.t_0 \; \dot{\mathbb{E}} \; \lambda x.t'_1$, and for all $v_0 \; \dot{\mathbb{E}} \; v_1$, we have $t_0\{v_0/x\} \; \dot{\mathbb{E}}\mathbb{E} \; t_1\{v_1/x\}$.*

Proof. We proceed by case analysis on $\lambda x.t_0 \; \dot{\mathbb{E}} \; t_1$.

Suppose $\lambda x.t_0 \mathbin{\mathbb{E}} t_1$. By definition, there exists t'_1 such that $t_1 \Downarrow_{\mathrm{v}} \lambda x.t'_1$ and $\{\lambda x.t_0, \lambda x.t'_1\} \in \mathbb{E}$. Because $\Downarrow_{\mathrm{v}} \subseteq \mathbb{E}$, we have $\lambda x.t_0 \mathbin{\mathbb{E}} \lambda x.t'_1$. Let $v_0 \mathbin{\mathbb{E}} v_1$. By Lemma C.1, we have $t_0\{v_0/x\} \mathbin{\dot{\mathbb{E}}} t_0\{v_1/x\}$, and by Lemma C.5, we have $t_0\{v_1/x\} \mathbin{\mathbb{E}} t'_1\{v_1/x\}$, hence the result holds.

Suppose $t_0 \mathbin{\dot{\mathbb{E}}} t_1$ with $\mathsf{fv}(t_0) \cup \mathsf{fv}(t) \subseteq \{x\}$. We have $t_0\{v_0/x\} \mathbin{\dot{\mathbb{E}}} t_1\{v_1/x\}$ by Lemma C.1, hence the result holds.

Lemma C.7. *If $E_0[\mathcal{S}k.t_0] \mathbin{\dot{\mathbb{E}}} t_1$, then there exists $E_1[\mathcal{S}k.t'_1]$ such that $t_1 \Downarrow_{\mathrm{v}} E_1[\mathcal{S}k.t'_1]$, and for all $E'_0 \mathbin{\ddot{\mathbb{E}}} E'_1$, we have*

$$\langle t_0\{\lambda x.\langle E'_0[E_0[x]]\rangle/k\}\rangle \mathbin{\dot{\mathbb{E}}\mathbb{E}} \langle t'_1\{\lambda x.\langle E'_1[E_1[x]]\rangle/k\}\rangle.$$

Proof. We proceed by induction on $E_0[\mathcal{S}k.t_0] \mathbin{\dot{\mathbb{E}}} t_1$. If $E_0[\mathcal{S}k.t_0] \mathbin{\mathbb{E}} t_1$, then there exists $E_1[\mathcal{S}k.t'_1]$ such that $t_1 \Downarrow_{\mathrm{v}} E_1[\mathcal{S}k.t'_1]$, and for all context E'_1, we have $\langle t_0\{\lambda x.\langle E'_1[E_0[x]]\rangle/k\}\rangle \mathbin{\mathbb{E}} \langle t_1\{\lambda x.\langle E'_1[E_1[x]]\rangle/k\}\rangle$ (*). Let $E'_0 \mathbin{\ddot{\mathbb{E}}} E'_1$; then we have $\lambda x.\langle E'_0[E_0[x]]\rangle \mathbin{\dot{\mathbb{E}}} \lambda x.\langle E'_1[E_0[x]]\rangle$. Lemma C.1 gives us $\langle t_0\{\lambda x.\langle E'_0[E_0[x]]\rangle/k\}\rangle \mathbin{\dot{\mathbb{E}}} \langle t_0\{\lambda x.\langle E'_1[E_0[x]]\rangle/k\}\rangle$, which used with (*) implies $\langle t_0\{\lambda x.\langle E'_0[E_0[x]]\rangle/k\}\rangle \mathbin{\dot{\mathbb{E}}\mathbb{E}} \langle t_1\{\lambda x.\langle E'_1[E_1[x]]\rangle/k\}\rangle$, as required.

Suppose $E_0 = E = \square$ and $t_0 \mathbin{\dot{\mathbb{E}}} t'_1$ with $\mathsf{fv}(t_0) \cup \mathsf{fv}(t) \subseteq \{k\}$. From $E'_0 \mathbin{\ddot{\mathbb{E}}} E'_1$, we have $\lambda x.\langle E'_0[x]\rangle \mathbin{\dot{\mathbb{E}}} \lambda x.\langle E'_1[x]\rangle$. Then $\langle t_0\{\lambda x.\langle E'_0[x]\rangle/k\}\rangle \mathbin{\dot{\mathbb{E}}} \langle t'_1\{\lambda x.\langle E'_1[x]\rangle/k\}\rangle$ holds by Lemma C.1, hence the result holds.

Suppose $E_0[\mathcal{S}k.t_0] = v_0\, E''_0[\mathcal{S}k.t_0]$, $t_1 = t_1^1 t_1^2$ with $v_0 \mathbin{\dot{\mathbb{E}}} t_1^1$ and $E''_0[\mathcal{S}k.t_0] \mathbin{\dot{\mathbb{E}}} t_1^2$. By Lemma C.6, there exists v_1 such that $t_1^1 \Downarrow_{\mathrm{v}} v_1$ and $v_0 \mathbin{\dot{\mathbb{E}}} v_1$. Let $E'_0 \mathbin{\ddot{\mathbb{E}}} E'_1$; then we have $E'_0[v_0\ \square] \mathbin{\ddot{\mathbb{E}}} E'_1[v_1\ \square]$. By the induction hypothesis, there exists $E''_1[\mathcal{S}k.t'_1]$ such that $t_1^2 \Downarrow_{\mathrm{v}} E''_1[\mathcal{S}k.t'_1]$ and

$$\langle t_0\{\lambda x.\langle E'_0[v_0\ E''_0[x]]\rangle/k\}\rangle \mathbin{\dot{\mathbb{E}}\mathbb{E}} \langle t'_1\{\lambda x.\langle E'_1[v_1\ E''_1[x]]\rangle/k\}\rangle.$$

Because $t_1 \Downarrow_{\mathrm{v}} v_1\, E''_1[\mathcal{S}k.t'_1]$, we have the required result. The case $E_0[\mathcal{S}k.t_0] = E''_0[\mathcal{S}k.t_0]\, t'_0$ is treated similarly.

Lemma C.8. *$t_0 \mathbin{\mathbb{E}} t_1$ implies $C[t_0] \approx_{\dot{\mathbb{E}}^{\mathrm{v}}} C[t_1]$.*

Proof. We prove that

$$\mathcal{X} = \{(\dot{\mathbb{E}}^{\mathrm{v}}, t_0, t_1) \mid t_0 \mathbin{\dot{\mathbb{E}}} t_1\} \cup \{\dot{\mathbb{E}}^{\mathrm{v}}\}$$

is a bisimulation up-to bisimilarity. Note that by definition of $\mathcal{X}$, we have $t \mathrel{\mathcal{X}_{\dot{\mathbb{E}}^{\mathrm{v}}}} t'$ iff $t \mathbin{\dot{\mathbb{E}}} t'$. Let $t_0 \mathrel{\mathcal{X}_{\dot{\mathbb{E}}^{\mathrm{v}}}} t_1$. We prove the bisimulation clauses by induction on $t_0 \mathbin{\dot{\mathbb{E}}} t_1$. The case $t_0 \mathbin{\mathbb{E}} t_1$ holds because $\mathbb{E}$ is an environmental bisimulation.

Suppose $t_0 = \lambda x.t'_0$, $t_1 = \lambda x.t'_1$ with $t'_0 \mathbin{\dot{\mathbb{E}}} t'_1$ and $\mathsf{fv}(t'_0) \cup \mathsf{fv}(t'_1) \subseteq \{x\}$. We have to prove that $(\dot{\mathbb{E}}^{\mathrm{v}} \cup \{(t_0, t_1)\}) \in \mathcal{X}$, i.e., $\dot{\mathbb{E}}^{\mathrm{v}} \in \mathcal{X}$, which is true.

Suppose $t_0 = t_0^1 t_0^2$, $t_1 = t_1^1 t_0^2$ with $t_0^1 \mathbin{\dot{\mathbb{E}}} t_1^1$ and $t_0^2 \mathbin{\dot{\mathbb{E}}} t_1^2$. We distinguish several cases.

- If $t_0^1 \rightarrow_{\mathrm{v}} {t_0^1}'$, then $t_0 \rightarrow_{\mathrm{v}} {t_0^1}'\, t_0^2$. By induction there exist ${t_1^1}''$, ${t_1^1}'$ such that $t_1^1 \rightarrow_{\mathrm{v}}^* {t_1^1}'$ and ${t_0^1}' \mathbin{\dot{\mathbb{E}}} {t_1^1}'' \mathbin{\mathbb{E}} {t_1^1}'$. Consequently we have $t_1 \rightarrow_{\mathrm{v}}^* {t_1^1}'\, t_1^2$. By definition of $\dot{\mathbb{E}}$, we have ${t_0^1}'\, t_0^2 \mathbin{\dot{\mathbb{E}}} {t_1^1}''\, t_1^2$, and by Lemma C.3, we have ${t_1^1}''\, t_1^2 \mathbin{\mathbb{E}} {t_1^1}'\, t_1^2$, hence ${t_0^1}'\, t_0^2 \mathbin{\dot{\mathbb{E}}\mathbb{E}} {t_1^1}'\, t_1^2$ holds, as wished.

- If t_0^1 is a value v_0 and $t_0^2 \rightarrow_{\mathrm{v}} t_0^{2'}$, then $t_0 \rightarrow_{\mathrm{v}} v_0\, t_0^{2'}$. By induction there exist $t_1^{2''}$, $t_1^{2'}$ such that $t_1^2 \rightarrow_{\mathrm{v}}^* t_1^{2'}$ and $t_0^{2'} \mathrel{\dot{\mathbb{E}}} t_1^{2''} \mathrel{\mathbb{E}} t_1^{2'}$. By Lemma C.6, there exists v_1 such that $t_1^1 \rightarrow_{\mathrm{v}}^* v_1$ and $v_0 \mathrel{\dot{\mathbb{E}}} v_1$. Consequently we have $t_1 \rightarrow_{\mathrm{v}}^* v_1 t_1^{2'}$. By definition of $\dot{\mathbb{E}}$, we have $v_0 t_0^{2'} \mathrel{\dot{\mathbb{E}}} v_1 t_1^{2''}$, and by Lemma C.3, we have $v_1\, t_1^{2''} \mathrel{\mathbb{E}} v_1\, t_1^{2'}$, hence $v_0\, t_0^{2'} \mathrel{\dot{\mathbb{E}}\mathbb{E}} v_1\, t_1^{2'}$ holds, as wished.

- If $t_0^1 = \lambda x.t_0'$ and $t_0^2 \rightarrow_{\mathrm{v}} v_0$, then $t_0 \rightarrow_{\mathrm{v}} t_0'\{v_0/x\}$. By Lemma C.6, there exist t_1', v_1 such that $t_1^1 \rightarrow_{\mathrm{v}}^* \lambda x.t_1'$, $\lambda x.t_0' \mathrel{\dot{\mathbb{E}}} \lambda x.t_1'$, $t_1^2 \rightarrow_{\mathrm{v}}^* v_1$, and $v_0 \mathrel{\dot{\mathbb{E}}} v_1$, which implies $t_0'\{v_0/x\} \mathrel{\dot{\mathbb{E}}\mathbb{E}} t_1'\{v_1/x\}$ by the same lemma. Because we also have $t_1 \rightarrow_{\mathrm{v}}^* t_1'\{v_1/x\}$, the result holds.

- If $t_0 = E_0[\mathcal{S}k.t_0']\, t_0^2$, then by Lemma C.7, there exists $E_1[\mathcal{S}k.t_1']$ such that $t_1^1 \rightarrow_{\mathrm{v}}^* E_1[\mathcal{S}k.t_1']$. Let $E_0'\ (\dot{\ddot{\mathbb{E}}}^{\mathrm{v}})\ E_1'$; then $E_0'[\Box\ t_0^2] \mathrel{\ddot{\mathbb{E}}} E_1'[\Box\ t_1^2]$ holds. By Lemma C.7, we have $\langle t_0'\{\lambda x.\langle E_0'[E_0[x]\, t_0^2]\rangle/k\}\rangle \mathrel{\dot{\mathbb{E}}\mathbb{E}} \langle t_1'\{\lambda x.\langle E_1'[E_1[x]\, t_1^2]\rangle/k\}\rangle$. Because $t_1 \rightarrow_{\mathrm{v}}^* E_1[\mathcal{S}k.t_1']\, t_1^2$, the result holds. The reasoning is the same if $t_0 = v_0\, E_0[\mathcal{S}k.t_0']$.

Suppose $t_0 = \langle t_0' \rangle$ and $t_1 = \langle t_1' \rangle$ with $t_0' \mathrel{\dot{\mathbb{E}}} t_1'$. We have three cases to consider.

- If $t_0' \rightarrow_{\mathrm{v}} t_0''$, then $t_0 \rightarrow_{\mathrm{v}} \langle t_0'' \rangle$. By induction there exists t_1'' such that $t_1' \rightarrow_{\mathrm{v}}^* t_1''$ and $t_0'' \mathrel{\dot{\mathbb{E}}\mathbb{E}} t_1''$. Consequently we have $t_1 \rightarrow_{\mathrm{v}}^* \langle t_1'' \rangle$, and by definition of $\dot{\mathbb{E}}$ and Lemma C.3, we have $\langle t_0'' \rangle \mathrel{\dot{\mathbb{E}}\mathbb{E}} \langle t_1'' \rangle$.

- If $t_0' = E_0[\mathcal{S}k.t_0'']$, then $t_0 \rightarrow_{\mathrm{v}} \langle t_0''\{\lambda x.\langle E_0[x]\rangle/k\}\rangle$. By Lemma C.7, there exist E_1 and t_1'' such that $t_1' \rightarrow_{\mathrm{v}}^* E_1[\mathcal{S}k.t_1'']$ and $\langle t_0''\{\lambda x.\langle E_0[x]\rangle/k\}\rangle \mathrel{\dot{\mathbb{E}}\mathbb{E}} \langle t_1''\{\lambda x.\langle E_1[x]\rangle/k\}\rangle$, as wished.

- If $t_0' = v_0$, then $t_0 \rightarrow_{\mathrm{v}} v_0$. By Lemma C.6, there exists v_1 such that $t_1' \rightarrow_{\mathrm{v}}^* v_1$ and $v_0 \mathrel{\dot{\mathbb{E}}} v_1$. We have $t_1 \rightarrow_{\mathrm{v}}^* v_1$, hence the result holds.

Suppose $t_0 = \mathcal{S}k.t_0'$ and $t_1 = \mathcal{S}k.t_1'$ with $t_0' \mathrel{\dot{\mathbb{E}}} t_1'$ and $\mathsf{fv}(t_0') \cup \mathsf{fv}(t_1') \subseteq \{k\}$. By compatibility of $\dot{\mathbb{E}}$, we have $\langle t_0' \rangle \mathrel{\dot{\mathbb{E}}} \langle t_1' \rangle$ Let $E_0\ (\dot{\ddot{\mathbb{E}}}^{\mathrm{v}})\ E_1$; we have $\lambda x.\langle E_0[x]\rangle \mathrel{\dot{\mathbb{E}}} \lambda x.\langle E_1[x]\rangle$. By Lemma C.1, we have $\langle t_0'\{\lambda x.\langle E_0[x]\rangle/k\}\rangle \mathrel{\dot{\mathbb{E}}} \langle t_1'\{\lambda x.\langle E_1[x]\rangle/k\}\rangle$, as wished.

We now verify the conditions on the environment $\dot{\mathbb{E}}$. Suppose $\lambda x.t_0 \mathrel{\dot{\mathbb{E}}} \lambda x.t_1$ and $v_0 \mathrel{\dot{\mathbb{E}}} v_1$. Then by Lemma C.6 and reflexivity of $\mathbb{E}$, we have $t_0\{v_0/x\} \mathrel{\mathbb{E}\dot{\mathbb{E}}\mathbb{E}} t_1\{v_0/x\}$, as wished.

Corollary C.9. *For all $\mathcal{E} \in \approx$, $\approx_{\mathcal{E}}$ is a congruence.*

Proof. If $t_0 \approx_{\mathcal{E}} t_1$, then by weakening (Proposition 6.5), we have $t_0 \mathrel{\mathbb{E}} t_1$, which in turn implies $C[t_0] \approx_{\dot{\mathbb{E}}^{\mathrm{v}}} C[t_1]$ (by Lemma C.8), and gives us $C[t_0] \approx_{\mathcal{E}} C[t_1]$ using weakening again.

C.2. Soundness Proof for the Original Semantics

The proof scheme is the same as for the relaxed semantics. We point out the main differences between the two proofs.

Lemma C.10. *Let $\mathcal{E}$ be an environment*

- *If $v_0 \approx_{\mathcal{E}} v_1$, then $C[v_0] \simeq_{\dot{\mathcal{E}}^{\mathrm{v}}} C[v_1]$.*
- *If $t_0 \approx_{\mathcal{E}} t_1$, then $F[t_0] \simeq_{\dot{\mathcal{E}}^{\mathrm{v}}} F[t_1]$.*

Let $\mathcal{Y}$ be an environmental bisimulation. We define

$$\begin{aligned}
\mathcal{X} &= \mathcal{X}_1 \cup \mathcal{X}_2 \cup \{\dot{\mathcal{E}}^{\mathrm{v}} \mid \mathcal{E} \in \mathcal{Y}\}\\
\mathcal{X}_1 &= \{(\dot{\mathcal{E}}^{\mathrm{v}}, F_0[t_0], F_1[t_1]) \mid t_0 \ \mathcal{Y}_{\mathcal{E}}\ t_1, F_0 \ \ddot{\mathcal{E}}\ F_1\}\\
\mathcal{X}_2 &= \{(\dot{\mathcal{E}}^{\mathrm{v}}, t_0, t_1) \mid \mathcal{E} \in \mathcal{Y}, t_0 \ \dot{\mathcal{E}}\ t_1\}
\end{aligned}$$

Lemma C.11. *Let $\mathcal{E} \in \mathcal{Y}$. If $\lambda x.t_0 \ \dot{\mathcal{E}}\ \lambda x.t_1$ and $v_0 \ \dot{\mathcal{E}}^{\mathrm{v}}\ v_1$ then $t_0\{v_0/x\} \ \mathcal{X}_{\dot{\mathcal{E}}^{\mathrm{v}}}\ t_1\{v_1/x\}$.*

Proof. Similar to the proof of Lemma C.4.

We prove Lemma C.10 by showing that $\mathcal{X}$ is a bisimulation up to environment.

Proof. We first prove the bisimulation for the elements in $\mathcal{X}_2$. Let $t_0 \ \dot{\mathcal{E}}\ t_1$, with $\mathcal{E} \in \mathcal{Y}$. If t_0 or t_1 is not delimited, then for all $E_0 \ \ddot{\mathcal{E}}\ E_1$, we have $\langle E_0[t_0]\rangle \ \dot{\mathcal{E}}\ \langle E_1[t_1]\rangle$, i.e., $\langle E_0[t_0]\rangle \ \mathcal{X}_{\dot{\mathcal{E}}^{\mathrm{v}}}\ \langle E_1[t_1]\rangle$, as required.

Suppose t_0, t_1 are delimited terms p_0, p_1. If $p_0 \rightarrow_{\mathrm{v}} v_0$, then $p_0 = \langle v_0\rangle$, and therefore $p_1 = \langle v_1\rangle$ with $v_0 \ \dot{\mathcal{X}}\ v_1$. We have $p_1 \rightarrow_{\mathrm{v}} v_1$, and also $\{(v_0, v_1)\} \cup \dot{\mathcal{E}}^{\mathrm{v}} = \dot{\mathcal{E}}^{\mathrm{v}} \in \mathcal{X}$, as wished.

Otherwise $p_0 \rightarrow_{\mathrm{v}} p_0'$. Then $p_0 = F_0[r_0]$. Because $\mathcal{E}$ relates only values, we can prove there exist F_1, r_1 such that $p_1 = F_1[r_1]$, $F_0 \ \ddot{\mathcal{E}}\ F_1$, and $r_0 \ \dot{\mathcal{E}}\ r_1$. We distinguish several cases, depending on the redexes.

- If $r_0 = \langle v_0\rangle$ and $r_1 = \langle v_1\rangle$ with $v_0 \ \dot{\mathcal{E}}^{\mathrm{v}}\ v_1$, then $p_0 \rightarrow_{\mathrm{v}} F_0[v_0]$ and $p_1 \rightarrow_{\mathrm{v}} F_1[v_1]$. We have $F_0[v_0] \ \dot{\mathcal{E}}\ F_1[v_1]$, as wished.
- Suppose $r_0 = (\lambda x.t_0')\, v_0$ and $r_1 = (\lambda x.t_1')\, v_1$ with $\lambda x.t_0' \ \dot{\mathcal{E}}^{\mathrm{v}}\ \lambda x.t_1'$ and $v_0 \ \dot{\mathcal{E}}^{\mathrm{v}}\ v_1$. Then $p_0 \rightarrow_{\mathrm{v}} F_0[t_0'\{v_0/x\}]$ and $p_1 \rightarrow_{\mathrm{v}} F_1[t_1'\{v_1/x\}]$. By Lemma C.11 and because $\mathcal{X}$ is compatible, we have $F_0[t_0'\{v_0/x\}] \ \mathcal{X}_{\dot{\mathcal{E}}^{\mathrm{v}}}\ F_1[t_1'\{v_1/x\}]$, as wished.
- If $r_0 = \langle E_0[\mathcal{S}k.t_0']\rangle$ and $r_1 = \langle E_1[\mathcal{S}k.t_1']\rangle$ with $E_0 \ \ddot{\mathcal{E}}\ E_1$ and $t_0' \ \dot{\mathcal{E}}^{\mathrm{v}}\ t_1'$. Then $p_0 \rightarrow_{\mathrm{v}} F_0[\langle t_0'\{\lambda x.\langle E_0[x]\rangle/k\}\rangle]$ and $p_1 \rightarrow_{\mathrm{v}} F_1[\langle t_1'\{\lambda x.\langle E_1[x]\rangle/k\}\rangle]$. From $E_0 \ \ddot{\mathcal{E}}\ E_1$, we deduce $\lambda x.\langle E_0[x]\rangle \ \dot{\mathcal{E}}^{\mathrm{v}}\ \lambda x.\langle E_1[x]\rangle$, so by Lemma C.1, we have $F_0[\langle t_0'\{\lambda x.\langle E_0[x]\rangle/k\}\rangle] \ \dot{\mathcal{E}}\ F_1[\langle t_1'\{\lambda x.\langle E_1[x]\rangle/k\}\rangle]$, as wished.

We now prove the bisimulation property (up to environment) for elements in $\mathcal{X}_1$. Let $F_0[t_0] \; \mathcal{X}_{\dot{\mathcal{E}}^{\mathrm{v}}} \; F_1[t_1]$, so that $t_0 \; \mathcal{Y}_{\mathcal{E}} \; t_1$ and $F_0 \; \ddot{\mathcal{E}} \; F_1$. If $F_0[t_0]$ and $F_1[t_1]$ are not both delimited terms, then for all $E_0 \; \ddot{\mathcal{E}} \; E_1$, we have $\langle E_0[F_0[t_0]]\rangle \; \mathcal{X}_{\dot{\mathcal{E}}^{\mathrm{v}}} \; \langle E_1[F_1[t_1]]\rangle$, as wished.

Suppose $F_0[t_0]$, $F_1[t_1]$ are delimited terms p_0, p_1. We distinguish two cases. First, suppose t_0 and t_1 are themselves delimited terms. If $t_0 \to_{\mathrm{v}} p_0'$, then $p_0 \to_{\mathrm{v}} F_0[p_0']$. Because $t_0 \; \mathcal{Y}_{\mathcal{E}} \; t_1$, there exists p_1' such that $t_1 \to_{\mathrm{v}}^* p_1'$ and $p_0' \; \mathcal{Y}_{\mathcal{E}} \; p_1'$. We have $F_0[p_0'] \; \mathcal{X}_{\dot{\mathcal{E}}^{\mathrm{v}}} \; F_1[p_1']$ and $p_1 \to_{\mathrm{v}}^* F_1[p_1']$, as wished. If $t_0 \to_{\mathrm{v}} v_0$, then $p_0 \to_{\mathrm{v}} F_0[v_0]$. Because $t_0 \; \mathcal{Y}_{\mathcal{E}} \; t_1$, there exists v_1 such that $t_1 \to_{\mathrm{v}}^* v_1$ and $\mathcal{E}' \stackrel{\mathsf{def}}{=} \{(v_0, v_1)\} \cup \mathcal{E} \in \mathcal{Y}$. Hence we have $p_1 \to_{\mathrm{v}}^* F_1[v_1]$ and $F_0[v_0] \; \mathcal{X}_{\dot{\mathcal{E}'}^{\mathrm{v}}} \; F_1[v_1]$, as wished.

In the second case, t_0 and t_1 are not both delimited terms. Then we can write $p_0 = F_0'[\langle E_0[t_0]\rangle]$ and $p_1 = F_1'[\langle E_1[t_1]\rangle]$ for some $F_0' \; \ddot{\mathcal{E}} \; F_1'$ and $E_0 \; \ddot{\mathcal{E}} \; E_1$. Because $t_0 \; \mathcal{Y}_{\mathcal{E}} \; t_1$ and since $\mathcal{Y}$ is an environmental bisimulation, we have $\langle E_0[t_0]\rangle \; \mathcal{Y}_{\mathcal{E}} \; \langle E_1[t_1]\rangle$. If $\langle E_0[t_0]\rangle \to_{\mathrm{v}} p_0'$, then there exists p_1' such that $\langle E_1[t_1]\rangle \to_{\mathrm{v}}^* p_1'$ and $p_0' \; \mathcal{Y}_{\mathcal{E}} \; p_1'$. Therefore, $p_0 \to_{\mathrm{v}} F_0'[p_0']$, $p_1 \to_{\mathrm{v}} F_1'[p_1']$, and $F_0'[p_0'] \; \mathcal{X}_{\dot{\mathcal{E}}^{\mathrm{v}}} \; F_1'[p_1']$ hold, as wished. If $\langle E_0[t_0]\rangle \to_{\mathrm{v}} v_0$, then there exists v_1 such that $\langle E_1[t_1]\rangle \to_{\mathrm{v}}^* v_1$ and $\mathcal{E}' \stackrel{\mathsf{def}}{=} \{(v_0, v_1)\} \cup \mathcal{E} \subseteq \mathcal{Y}$. Therefore we have $p_0 \to_{\mathrm{v}} F_0'[v_0]$, $p_1 \to_{\mathrm{v}} F_1'[v_1]$, and $F_0'[v_0] \; \mathcal{X}_{\dot{\mathcal{E}'}^{\mathrm{v}}} \; F_1'[v_1]$, as wished.

Finally we prove the clause about environments. Let $\lambda x.t_0 \; \dot{\mathcal{E}}^{\mathrm{v}} \; \lambda x.t_1$ and $v_0 \; \dot{\mathcal{E}}^{\mathrm{v}} \; v_1$. By Lemma C.11, we have $t_0\{v_0/x\} \; \mathcal{X}_{\dot{\mathcal{E}}^{\mathrm{v}}} \; t_1\{v_1/x\}$, as wished.

Lemma C.12. *If $\lambda x.t_0 \; \mathbb{F} \; \lambda x.t_1$, then $t_0\{v/x\} \; \mathbb{F} \; t_1\{v/x\}$.*

Proof. It follows from the definition of the bisimulation and weakening.

Lemma C.13. *If $\lambda x.t_0 \; \dot{\mathbb{F}} \; \lambda x.t \; \mathbb{F} \; \lambda x.t_1$ and $v_0 \; \dot{\mathbb{F}} \; v \; \mathbb{F} \; v_1$ then $t_0\{v_0/x\} \; \dot{\mathbb{F}}\mathbb{F} \; t_1\{v_1/x\}$.*

Proof. Similar to the proof of Lemma C.6.

Lemma C.14. *If $p_0 \; \dot{\mathbb{F}} \; p_1$ then we have one of the following cases:*

- $p_0 \; \mathbb{F} \; p_1$;
- $p_0 = \langle v_0 \rangle$;
- $p_0 = F_0[\langle E_0[t_0]\rangle]$, $p_1 = F_1[\langle E_1[t_1]\rangle]$, $F_0 \; \ddot{\mathbb{F}} \; F_1$, $E_0 \; \ddot{\mathbb{F}} \; E_1$, $t_0 \; \mathbb{F} \; t_1$ *and* $t_0 \to_{\mathrm{v}} t_0'$ *or* t_0 *is stuck;*
- $p_0 = F_0[\langle E_0[r_0]\rangle]$, $p_1 = F_1[\langle E_1[t_1]\rangle]$, $F_0 \; \ddot{\mathbb{F}} \; F_1$, $E_0 \; \ddot{\mathbb{F}} \; E_1$, $r_0 \; \dot{\mathbb{F}} \; t_1$ *but* $r_0 \; \not\mathbb{F} \; t_1$.

Proof. We prove a more general result on $t_0 \; \dot{\mathbb{F}} \; t_1$. We have either

- $t_0 \; \mathbb{F} \; t_1$;
- $t_0 = v_0$;

- $t_0 = E_0[t'_0]$, $t_1 = E_1[t'_1]$, $E_0 \ddot{\mathbb{F}} E_1$, $t'_0 \mathbb{F} t'_1$, and $t'_0 \rightarrow_{\rm v} t''_0$ or t_0 is stuck;
- $t_0 = F_0[\langle E_0[t'_0]\rangle]$, $t_1 = F_1[\langle E_1[t'_1]\rangle]$, $F_0 \ddot{\mathbb{F}} F_1$, $E_0 \ddot{\mathbb{F}} E_1$, $t'_0 \mathbb{F} t'_1$, and $t'_0 \rightarrow_{\rm v} t''_0$ or t'_0 is stuck;
- $t_0 = E_0[r_0]$, $t_1 = E_1[t'_1]$, $E_0 \ddot{\mathbb{F}} E_1$, $r_0 \dot{\mathbb{F}} t'_1$ but $r_0 \not\mathbb{F} t'_1$.
- $t_0 = F_0[\langle E_0[r_0]\rangle]$, $t_1 = F_1[\langle E_1[t'_1]\rangle]$, $F_0 \ddot{\mathbb{F}} F_1$, $E_0 \ddot{\mathbb{F}} E_1$, $r_0 \dot{\mathbb{F}} t'_1$ but $r_0 \not\mathbb{F} t'_1$.

The proof is straightforward by induction on $t_0 \dot{\mathbb{F}} t_1$.

Lemma C.15. *If $v_0 \dot{\mathbb{F}} t_1$, then there exists v_1 such that $\langle t_1\rangle \rightarrow^*_{\rm v} v_1$ and $v_0 \dot{\mathbb{F}} v_1$.*

Proof. If $v_0 \mathbb{F} t_1$, then we can conclude using the definition of the bisimilarity. Otherwise, t_1 is a value v_1, and the result holds trivially.

Lemma C.16. *Let $t_0 \mathbb{F} t_1$ so that $t_0 \rightarrow_{\rm v} t'_0$ or t_0 is stuck, and $E_0 \ddot{\mathbb{F}} E_1$. There exist p'_0, p'_1 such that $\langle E_0[t_0]\rangle \rightarrow_{\rm v} p'_0$, $\langle E_1[t_1]\rangle \rightarrow^*_{\rm v} p'_1$, and $p'_0 \dot{\mathbb{F}}\mathbb{F} p'_1$.*

Proof. If t_0 and t_1 are both delimited terms, then t_0 cannot be stuck, and we conclude using bisimilarity and the definition of $\dot{\mathbb{F}}$.

Suppose t_0 and t_1 are not both delimited terms. Because $t_0 \rightarrow_{\rm v} t'_0$ or t_0 is stuck, we know there exist p'_0, p''_0 such that $\langle E_0[t_0]\rangle \rightarrow_{\rm v} p'_0$, $\langle E_1[t_0]\rangle \rightarrow_{\rm v} p''_0$, and $p'_0 \dot{\mathbb{F}} p''_0$. Because $t_0 \mathbb{F} t_1$, we have $\langle E_1[t_0]\rangle \mathbb{F} \langle E_1[t_1]\rangle$, and there exists p'_1 such that $\langle E_1[t_1]\rangle \rightarrow^*_{\rm v} p'_1$ and $p''_0 \mathbb{F} p'_1$. We have $p'_0 \dot{\mathbb{F}}\mathbb{F} p'_1$, hence the result holds.

Lemma C.17. *Let $\lambda x.t_0 \dot{\mathbb{F}} t^1_1$, $v_0 \dot{\mathbb{F}} t^2_1$, and $E_0 \ddot{\mathbb{F}} E_1$. There exist p_0, p_1 such that $\langle E_0[(\lambda x.t_0)\, v_0]\rangle \rightarrow_{\rm v} p_0$, $\langle E_1[t^1_1\, t^2_1]\rangle \rightarrow^*_{\rm v} p_1$, and $p_0 \dot{\mathbb{F}}\mathbb{F} p_1$.*

Proof. We have $\langle E_0[(\lambda x.t_0)\, v_0]\rangle \rightarrow_{\rm v} \langle E_0[t_0\{v_0/x\}]\rangle \stackrel{\mathsf{def}}{=} p_0$. To get the result of the reduction of $\langle E_1[t^1_1\, t^2_1]\rangle$, we distinguish several cases.

First, suppose $\lambda x.t_0 \mathbb{F} \lambda x.t_1$ with $t_0 \dot{\mathbb{F}} t_1$. If $v_0 \dot{\mathbb{F}} v_1$, then $t_0\{v_0/x\} \dot{\mathbb{F}}\mathbb{F} t_1\{v_1/x\}$ holds by Lemma C.13. We have $\langle E_1[(\lambda x.t_1)\, v_1]\rangle \rightarrow_{\rm v} \langle E_1[t_1\{v_1/x\}]\rangle$, and also $p_0 \dot{\mathbb{F}}\mathbb{F} \langle E_1[t_1\{t_1/x\}]\rangle$ by compatibility of $\dot{\mathbb{F}}$ and Lemma C.10. Therefore, we have the required result.

If $v_0 \mathbb{F} t^2_1$, then by bisimilarity, we have $\langle E_1[(\lambda x.t_1)\, v_0]\rangle \mathbb{F} \langle E_1[(\lambda x.t_1)\, t^2_1]\rangle$. Because $\langle E_1[(\lambda x.t_1)\, v_0]\rangle \rightarrow_{\rm v} \langle E_1[t_1\{v_0/x\}]\rangle$, by bisimilarity there exists p_1 such that $\langle E_1[(\lambda x.t_1)\, t^2_1]\rangle \rightarrow^*_{\rm v} p_1$ and $\langle E_1[t_1\{v_0/x\}]\rangle \mathbb{F} p_1$. We also have $p_0 \dot{\mathbb{F}} \langle E_1[t_1\{v_0/x\}]\rangle$, hence the result holds.

Next, suppose $\lambda x.t_0 \mathbb{F} t^1_1$. If $v_0 \dot{\mathbb{F}} v_1$, then we have $\langle E_1[(\lambda x.t_0)\, v_1]\rangle \mathbb{F} \langle E_1[t^1_1\, v_1]\rangle$ by bisimilarity. Therefore, there exists p_1 such that $\langle E_1[t^1_1\, v_1]\rangle \rightarrow^*_{\rm v} p_1$ and $\langle E_1[t_0\{v_1/x\}]\rangle \mathbb{F} p_1$. We also have $p_0 \dot{\mathbb{F}} \langle E_1[t_0\{v_1/x\}]\rangle$, hence the result holds.

If $v_0 \mathbb{F} t^2_1$, then by bisimilarity, we have $\langle E_1[(\lambda x.t_0)\, v_0]\rangle \mathbb{F} \langle E_1[t^1_1\, v_0]\rangle$ and $\langle E_1[t^1_1\, v_0]\rangle \mathbb{F} \langle E_1[t^1_1\, t^2_1]\rangle$. Therefore, there exists p'_1 such that $\langle E_1[t^1_1\, v_0]\rangle \rightarrow^*_{\rm v} p'_1$ and $\langle E_1[t_0\{v_0/x\}]\rangle \mathbb{F} p'_1$. It means that there exists p_1 such that $\langle E_1[t^1_1\, t^2_1]\rangle \rightarrow^*_{\rm v} p_1$ and $p'_1 \mathbb{F} p_1$. We also have $p_0 \dot{\mathbb{F}} \langle E_1[t_0\{v_0/x\}]\rangle$, hence the result holds.

Lemma C.18. *If $E[\mathcal{S}k.t_0] \mathrel{\dot{\mathbb{F}}} t_1$ and $E_0 \mathrel{\ddot{\mathbb{F}}} E_1$, then there exist p_0, p_1 such that $\langle E_0[E[\mathcal{S}k.t_0]]\rangle \rightarrow_{\mathrm{v}} p_0$, $\langle E_1[t_1]\rangle \rightarrow_{\mathrm{v}}^{*} p_1$, and $p_0 \mathrel{\dot{\mathbb{F}}\mathbb{F}} p_1$.*

Proof. There exists p_0 such that$\langle E_0[E[\mathcal{S}k.t_0]]\rangle \rightarrow_{\mathrm{v}} p_0$ by the capture reduction rule. To find p_1, we proceed by induction on $E[\mathcal{S}k.t_0] \mathrel{\dot{\mathbb{F}}} t_1$.

If $E[\mathcal{S}k.t_0] \mathrel{\mathbb{F}} t_1$, then by bisimilarity we have $\langle E_1[E[\mathcal{S}k.t_0]]\rangle \mathrel{\mathbb{F}} \langle E_1[t_1]\rangle$. Because $\langle E_1[E[\mathcal{S}k.t_0]]\rangle \rightarrow_{\mathrm{v}} p'_0$ for some p'_0, there exists p_1 such that $\langle E_1[t_1]\rangle \rightarrow_{\mathrm{v}}^{*} p_1$ and $p'_0 \mathrel{\mathbb{F}} p_1$. We also have $p_0 \mathrel{\dot{\mathbb{F}}} p'_0$, hence the result holds.

If $E = \square$, $t_1 = \mathcal{S}k.t'_1$ with $t_0 \mathrel{\dot{\mathbb{F}}} t'_1$, then we have $p_0 = \langle t_0\{\lambda k.\langle E_0[x]\rangle/k\}\rangle \mathrel{\dot{\mathbb{F}}} \langle t'_1\{\lambda k.\langle E_1[x]\rangle/k\}\rangle$ by Lemma C.1. Because $\langle E_1[\mathcal{S}k.t_1]\rangle \rightarrow_{\mathrm{v}} \langle t'_1\{\lambda k.\langle E_1[x]\rangle/k\}\rangle$, the required result holds.

Suppose $E[\mathcal{S}k.t_0] = v_0\, E'[\mathcal{S}k.t_0]$, $t_1 = t_1^1\, t_1^2$ with $v_0 \mathrel{\dot{\mathbb{F}}} t_1^1$ and $E'[\mathcal{S}k.t_0] \mathrel{\dot{\mathbb{F}}} t_1^2$. We distinguish two cases. If $t_1^1 = v_1$, then from $E_0 \mathrel{\ddot{\mathbb{F}}} E_1$ and $v_0 \mathrel{\dot{\mathbb{F}}} v_1$, we deduce $E_0[v_0\ \square] \mathrel{\ddot{\mathbb{F}}} E_1[v_1\ \square]$. We can apply the induction induction hypothesis with these two contexts, $E'[\mathcal{S}k.t_0]$, and t_1^2, and we obtain directly the required result. Suppose now $v_0 \mathrel{\mathbb{F}} t_1^1$. By the induction hypothesis (applied with $E_0[v_0\ \square]$, $E_1[v_0\ \square]$, $E'[\mathcal{S}k.t_0]$, and t_1^2), there exists p'_0 such that $\langle E_1[v_0\, t_1^2]\rangle \rightarrow_{\mathrm{v}}^{*} p'_0$, and $p_0 \mathrel{\dot{\mathbb{F}}\mathbb{F}} p'_0$. From $v_0 \mathrel{\mathbb{F}} t_1^1$, we know that $\langle E_1[v_0\, t_1^2]\rangle \mathrel{\mathbb{F}} \langle E_1[t_1^1\, t_1^2]\rangle$, which in turn implies that there exists p_1 such that $\langle E_1[t_1^1\, t_1^2]\rangle \rightarrow_{\mathrm{v}}^{*} p_1$ and $p'_0 \mathrel{\mathbb{F}} p_1$. Therefore we have $p_0 \mathrel{\dot{\mathbb{F}}\mathbb{F}} p_1$, hence the result holds.

Suppose $E[\mathcal{S}k.t_0] = E'[\mathcal{S}k.t_0]\, t$, $t_1 = t_1^1\, t_1^2$ with $E'[\mathcal{S}k.t_0] \mathrel{\dot{\mathbb{F}}} t_1^1$ and $t \mathrel{\dot{\mathbb{F}}} t_1^2$. By the induction hypothesis (applied to $E_0[\square\ t]$, $E_1[\square\ t_1^2]$, $E'[\mathcal{S}k.t_0]$, and t_1^1), there exists p_1 such that $\langle E_1[t_1^1\, t_1^2]\rangle \rightarrow_{\mathrm{v}}^{*} p_1$, and $p_0 \mathrel{\dot{\mathbb{F}}\mathbb{F}} p_1$, as required.

Lemma C.19. *$t_0 \mathrel{\mathbb{F}} t_1$ implies $C[t_0] \simeq_{\dot{\mathbb{F}}^{\mathrm{v}}} C[t_1]$.*

Proof. We prove that

$$\mathcal{X} = \{(\dot{\mathbb{F}}^{\mathrm{v}}, t_0, t_1) \ \mid\ t_0 \mathrel{\dot{\mathbb{F}}} t_1\} \cup \{\dot{\mathbb{F}}^{\mathrm{v}}\}$$

is a bisimulation up to bisimilarity. By definition of $\mathcal{X}$, we have $t \mathrel{\mathcal{X}_{\dot{\mathbb{F}}^{\mathrm{v}}}} t'$ iff $t \mathrel{\dot{\mathbb{F}}} t'$.

Let $t_0 \mathrel{\mathcal{X}_{\dot{\mathbb{F}}^{\mathrm{v}}}} t_1$. Suppose t_0 and t_1 are not both delimited terms. Then for all $E_0 \mathrel{\ddot{\mathbb{F}}} E_1$, we have $\langle E_0[t_0]\rangle \mathrel{\mathcal{X}_{\dot{\mathbb{F}}^{\mathrm{v}}}} \langle E_1[t_1]\rangle$, as required.

Suppose t_0 and t_1 are both delimited terms p_0, p_1. By Lemma C.14, we have several possibilities. If $p_0 \mathrel{\mathbb{F}} p_1$, then the result holds trivially. Suppose $p_0 = \langle v_0\rangle$, $p_1 = \langle t'_1\rangle$ with $v_0 \mathrel{\dot{\mathbb{F}}} t'_1$. By Lemma C.15, there exists v_1 such that $\langle t'_1\rangle \rightarrow_{\mathrm{v}}^{*} v_1$ and $v_0 \mathrel{\dot{\mathbb{F}}} v_1$. We have $\{(v_0, v_1)\}\cup\, \dot{\mathbb{F}}^{\mathrm{v}} {=} \dot{\mathbb{F}}^{\mathrm{v}} \in \mathcal{X}$, hence the result holds.

Suppose $p_0 = F_0[\langle E_0[t'_0]\rangle]$ and $p_1 = F_1[\langle E_1[t'_1]\rangle]$, with $F_0 \mathrel{\ddot{\mathbb{F}}} F_1$, $E_0 \mathrel{\ddot{\mathbb{F}}} E_1$, $t'_0 \mathrel{\mathbb{F}} t'_1$, and $t_0 \rightarrow_{\mathrm{v}} t'_0$ or t_0 is stuck. By Lemma C.16, there exist p'_0, p'_1 such that $\langle E_0[t'_0]\rangle \rightarrow_{\mathrm{v}} p'_0$, $\langle E_1[t'_1]\rangle \rightarrow_{\mathrm{v}}^{*} p'_1$, and $p'_0 \mathrel{\dot{\mathbb{F}}\mathbb{F}} p'_1$. By definition of $\dot{\mathbb{F}}$ and Lemma C.10, we have $F_0[p'_0] \mathrel{\dot{\mathbb{F}}\mathbb{F}} F_1[p'_1]$. Moreover $p_0 \rightarrow_{\mathrm{v}} F_0[p'_0]$ and $p_1 \rightarrow_{\mathrm{v}}^{*} F_1[p'_1]$, hence the result holds.

The last possibility is $p_0 = F_0[\langle E_0[r_0]\rangle]$, $p_1 = F_1[\langle E_1[t'_1]\rangle]$, with $F_0 \mathrel{\ddot{\mathbb{F}}} F_1$, $E_0 \mathrel{\ddot{\mathbb{F}}} E_1$, $r_0 \mathrel{\dot{\mathbb{F}}} t'_1$, and $r_0 \mathrel{\not{\mathbb{F}}} t'_1$. We discuss the three possible redexes. If $r_0 = \langle v_0\rangle$, then $t'_1 = \langle t''_1\rangle$ with $v_0 \mathrel{\dot{\mathbb{F}}} t''_1$. By Lemma C.15, there exists v_1

such that $\langle t_1'' \rangle \to_{\mathrm{v}}^* v_1$ and $v_0 \mathrel{\dot{\mathbb{F}}} v_1$. Then we have $p_0 \to_{\mathrm{v}} F_0[\langle E_0[v_0]\rangle]$ and $p_1 \to_{\mathrm{v}}^* F_1[\langle E_1[v_1]\rangle]$ with $F_0[\langle E_0[v_0]\rangle] \mathrel{\dot{\mathbb{F}}} F_1[\langle E_1[v_1]\rangle]$, hence the result holds. If $r_0 = v_0^1\, v_0^2$, then $t_1' = t_1^1\, t_1^2$ with $v_0^1 \mathrel{\dot{\mathbb{F}}} t_1^1$, and $v_0^2 \mathrel{\dot{\mathbb{F}}} t_1^2$. By Lemma C.17, there exist p_0', p_1' such that $E_0[r_0] \to_{\mathrm{v}} p_0'$, $E_1[t_1'] \to_{\mathrm{v}}^* p_1'$, and $p_0' \mathrel{\dot{\mathbb{F}}\mathbb{F}} p_1'$. Therefore we have $p_0 \to_{\mathrm{v}} F_0[p_0']$, $p_1 \to_{\mathrm{v}}^* F_1[p_1']$, with $F_0[p_0'] \mathrel{\dot{\mathbb{F}}\mathbb{F}} F_1[p_1']$ (by Lemma C.10 and definition of $\dot{\mathbb{F}}$), hence the result holds. The last case is $r_0 = \langle E[\mathcal{S}k.t_0']\rangle$; then $t_1' = \langle t_1'' \rangle$ with $E[\mathcal{S}k.t_0'] \mathrel{\dot{\mathbb{F}}} t_1''$. By Lemma C.18, there exist p_0', p_1' such that $r_0 \to_{\mathrm{v}} p_0'$, $t_1' \to_{\mathrm{v}}^* p_1'$ and $p_0' \mathrel{\dot{\mathbb{F}}\mathbb{F}} p_1'$. Therefore we have $p_0 \to_{\mathrm{v}} F_0[\langle E_0[p_0']\rangle]$, $p_1 \to_{\mathrm{v}}^* F_1[\langle E_1[p_1']\rangle]$, with $F_0[\langle E_0[p_0']\rangle] \mathrel{\dot{\mathbb{F}}\mathbb{F}} F_1[\langle E_1[p_1']\rangle]$ (by Lemma C.10 and definition of $\dot{\mathbb{F}}$), hence the result holds.

Finally, we check the clause for environments. let $\lambda x.t_0 \mathrel{\dot{\mathbb{F}}} \lambda x.t_1$ and $v_0 \mathrel{\dot{\mathbb{F}}} v_1$. By Lemma C.13, we get $t_0\{v_0/x\} \mathrel{\dot{\mathbb{F}}\mathbb{F}} t_1\{v_1/x\}$, hence the required result holds.

C.3. Proof of Proposition 6.40

Let t_0 and t_1 such that $k \notin \mathsf{fv}(t_1)$. We want to show that $(\lambda x.\mathcal{S}k.t_0)\, t_1 \mathrel{\mathbb{F}} \mathcal{S}k.((\lambda x.t_0)\, t_1)$. To this end, we need to plug both terms in some context $\langle E\rangle$, and compare $\langle E[(\lambda x.\mathcal{S}k.t_0)\, t_1]\rangle$ with $\langle E[\mathcal{S}k.((\lambda x.t_0)\, t_1)]\rangle$. The second term reduces to $\langle(\lambda x.t_0\{\lambda y.\langle E[y]\rangle/k\})\, t_1\rangle$, so we in fact prove the following result.

Lemma C.20. *We have* $\langle E[(\lambda x.\mathcal{S}k.t_0)\, t_1]\rangle \mathrel{\mathbb{F}} \langle(\lambda x.t_0\{\lambda y.\langle E[y]\rangle/k\})\, t_1\rangle$.

Proof. To make the proof easier to follow, we introduce some notations. We write $\overrightarrow{\cdot}$ for a sequence of entities (e.g., $\overrightarrow{E}$ for a sequence of contexts). We write $\boldsymbol{E}$ for $E[(\lambda x.\mathcal{S}k.t_0)\,\Box]$ and $\boldsymbol{E'}$ for $(\lambda x.t_0\{\lambda y.\langle E[y]\rangle/k\})\,\Box$, so the problem becomes relating $\langle \boldsymbol{E}[t_1]\rangle$ and $\langle \boldsymbol{E'}[t_1]\rangle$.

Next, given a sequence $E_0 \dots E_i$ of contexts such that $\mathsf{fv}(E_j) \subseteq \{k_0 \dots k_{j-1}\}$ for all $0 \leq j \leq i$, we inductively define families of substitutions $\sigma_0^{\overrightarrow{E}} \dots \sigma_i^{\overrightarrow{E}}$, $\delta_0^{\overrightarrow{E}} \dots \delta_i^{\overrightarrow{E}}$ as follows:

$$\begin{array}{rcl}
\sigma_0^{\overrightarrow{E}} &=& \cdot\{\lambda y.\langle \boldsymbol{E}[E_0[y]]\rangle/k_0\}\\
\delta_0^{\overrightarrow{E}} &=& \cdot\{\lambda y.\langle \boldsymbol{E'}[E_0[y]]\rangle/k_0\}\\
\sigma_j^{\overrightarrow{E}} &=& \cdot\{\lambda y.\langle \boldsymbol{E}[E_j\sigma_0^{\overrightarrow{E}} \dots \sigma_{j-1}^{\overrightarrow{E}}[y]]\rangle/k_j\} \text{ if } j > 0\\
\delta_j^{\overrightarrow{E}} &=& \cdot\{\lambda y.\langle \boldsymbol{E'}[E_j\delta_0^{\overrightarrow{E}} \dots \delta_{j-1}^{\overrightarrow{E}}[y]]\rangle/k_j\} \text{ if } j > 0
\end{array}$$

Finally, given a term t and a sequence of contexts $F_0 \dots F_i$, we inductively define families of terms $s_0^{t,\overrightarrow{F}} \dots s_i^{t,\overrightarrow{F}}$, $u_0^{t,\overrightarrow{F}} \dots u_i^{t,\overrightarrow{F}}$ as follows:

$$\begin{array}{rclrcl}
s_0^{t,\overrightarrow{F}} &=& F_0[\langle \boldsymbol{E}[t]\rangle] & s_j^{t,\overrightarrow{F}} &=& F_j[\langle \boldsymbol{E}[s_{j-1}^{t,\overrightarrow{F}}]\rangle] \text{ if } j > 0\\
u_0^{t,\overrightarrow{F}} &=& F_0[\langle \boldsymbol{E'}[t]\rangle] & u_0^{t,\overrightarrow{F}} &=& F_j[\langle \boldsymbol{E'}[u_{j-1}^{t,\overrightarrow{F}}]\rangle] \text{ if } j > 0
\end{array}$$

Note that the term we want to relate are $s_0^{t_1,\Box}$ and $u_0^{t_1,\Box}$. We let $\mathcal{E}$ ranges over environments of the form $\{(v\sigma_0^{\overrightarrow{E}} \dots \sigma_i^{\overrightarrow{E}}, v\delta_0^{\overrightarrow{E}} \dots \delta_i^{\overrightarrow{E}}) \mid \mathsf{fv}(v) \subseteq \{k_0 \dots k_i\}\} \cup \{(k_j\sigma_j^{\overrightarrow{E}}, k_j\delta_j^{\overrightarrow{E}})\}$. We prove that the relation

$$\mathcal{X} = \{(\mathcal{E}, t\sigma_0^{\overrightarrow{E}} \dots \sigma_i^{\overrightarrow{E}}, t\delta_0^{\overrightarrow{E}} \dots \delta_i^{\overrightarrow{E}}) \mid \mathsf{fv}(t) \subseteq \{k_0 \dots k_i\}\} \cup$$
$$\{(\mathcal{E}, \langle s_i^{t,\overrightarrow{F}}\rangle\sigma_0^{\overrightarrow{E'}} \dots \sigma_j^{\overrightarrow{E'}}, \langle u_i^{t,\overrightarrow{F}}\rangle\delta_0^{\overrightarrow{E'}} \dots \delta_j^{\overrightarrow{E'}}) \mid \mathsf{fv}(t) \cup \mathsf{fv}(\overrightarrow{F}) \subseteq \{k_0 \dots k_j\}\} \cup \{\mathcal{E}\}$$

is a delimited bisimulation. Let $t\sigma_0^{\overrightarrow{E}} \dots \sigma_i^{\overrightarrow{E}} \ \mathcal{X}_{\mathcal{E}} \ t\delta_0^{\overrightarrow{E}} \dots \delta_i^{\overrightarrow{E}}$ where t is not delimited. Let $E_0 \ \ddot{\mathcal{E}} \ E_1$; by definition of $\mathcal{E}$, we have $E_0 = E'\sigma_0^{\overrightarrow{E''}} \dots \sigma_j^{\overrightarrow{E''}}$ and $E_1 = E'\delta_0^{\overrightarrow{E''}} \dots \delta_j^{\overrightarrow{E''}}$ for some E', $\overrightarrow{E''}$. With some renumbering and rewriting, we have $\langle E_0[t\sigma_0^{\overrightarrow{E}} \dots \sigma_i^{\overrightarrow{E}}]\rangle = \langle E'[t]\rangle\sigma_0^{\overrightarrow{E},\overrightarrow{E''}} \dots \sigma_{i+j+1}^{\overrightarrow{E},\overrightarrow{E''}}$ and $\langle E_1[t\sigma_0^{\overrightarrow{E}} \dots \sigma_i^{\overrightarrow{E}}]\rangle = \langle E'[t]\rangle\delta_0^{\overrightarrow{E},\overrightarrow{E''}} \dots \delta_{i+j+1}^{\overrightarrow{E},\overrightarrow{E''}}$: the two terms are in $\mathcal{X}$, as wished.

Let $\langle t\rangle\sigma_0^{\overrightarrow{E}} \dots \sigma_i^{\overrightarrow{E}} \ \mathcal{X}_{\mathcal{E}} \ \langle t\rangle\delta_0^{\overrightarrow{E}} \dots \delta_i^{\overrightarrow{E}}$. We have three cases for t.

If $\langle t\rangle \rightarrow_{\mathrm{v}} \langle t'\rangle$, we still have $\langle t'\rangle\sigma_0^{\overrightarrow{E}} \dots \sigma_i^{\overrightarrow{E}} \ \mathcal{X}_{\mathcal{E}} \ \langle t'\rangle\delta_0^{\overrightarrow{E}} \dots \delta_i^{\overrightarrow{E}}$. If $\langle t\rangle \rightarrow_{\mathrm{v}} v$ or $t = k_j$, then both terms reduce to values that are in $\mathcal{E}$, by definition of $\mathcal{E}$.

If $t = F[k_j\ v]$, then

$$\begin{array}{rcl}\langle t\rangle\sigma_0^{\overrightarrow{E}} \dots \sigma_i^{\overrightarrow{E}} & = & F\sigma_0^{\overrightarrow{E}} \dots \sigma_i^{\overrightarrow{E}}[\lambda y.\langle \boldsymbol{E}[E_j\sigma_0^{\overrightarrow{E}} \dots \sigma_{j-1}^{\overrightarrow{E}}[y]]\rangle \ v\sigma_0^{\overrightarrow{E}} \dots \sigma_i^{\overrightarrow{E}}] \\ \langle t\rangle\delta_0^{\overrightarrow{E}} \dots \delta_i^{\overrightarrow{E}} & = & F\delta_0^{\overrightarrow{E}} \dots \delta_i^{\overrightarrow{E}}[\lambda y.\langle \boldsymbol{E'}[E_j\delta_0^{\overrightarrow{E}} \dots \delta_{j-1}^{\overrightarrow{E}}[y]]\rangle \ v\delta_0^{\overrightarrow{E}} \dots \delta_i^{\overrightarrow{E}}]\end{array}$$

Reducing the β-redex in both terms, we obtain

$$\begin{array}{rcl}\langle t\rangle\sigma_0^{\overrightarrow{E}} \dots \sigma_i^{\overrightarrow{E}} & \rightarrow_{\mathrm{v}} & F\sigma_0^{\overrightarrow{E}} \dots \sigma_i^{\overrightarrow{E}}[\langle \boldsymbol{E}[E_j\sigma_0^{\overrightarrow{E}} \dots \sigma_{j-1}^{\overrightarrow{E}}[v\sigma_0^{\overrightarrow{E}} \dots \sigma_i^{\overrightarrow{E}}]]\rangle] \\ \langle t\rangle\delta_0^{\overrightarrow{E}} \dots \delta_i^{\overrightarrow{E}} & \rightarrow_{\mathrm{v}} & F\delta_0^{\overrightarrow{E}} \dots \delta_i^{\overrightarrow{E}}[\langle \boldsymbol{E'}[E_j\delta_0^{\overrightarrow{E}} \dots \delta_{j-1}^{\overrightarrow{E}}[v\delta_0^{\overrightarrow{E}} \dots \delta_i^{\overrightarrow{E}}]]\rangle]\end{array}$$

The resulting terms can be written $\langle s_0^{t',\overrightarrow{F}}\rangle\sigma_0^{\overrightarrow{E}} \dots \sigma_j^{\overrightarrow{E}}$ and $\langle u_0^{t',\overrightarrow{F}}\rangle\delta_0^{\overrightarrow{E}} \dots \delta_j^{\overrightarrow{E}}$, with $t' = E_j[v]$, therefore we obtain terms in $\mathcal{X}_{\mathcal{E}}$.

Let $\langle s_i^{t,\overrightarrow{F}}\rangle\sigma_0^{\overrightarrow{E'}} \dots \sigma_j^{\overrightarrow{E'}} \ \mathcal{X}_{\mathcal{E}} \ \langle u_i^{t,\overrightarrow{F}}\rangle\delta_0^{\overrightarrow{E'}} \dots \delta_j^{\overrightarrow{E'}}$. One can check that the reductions from terms of the form $s_i^{t,\overrightarrow{F}}$, $u_i^{t,\overrightarrow{F}}$ come from respectively $s_0^{t\overrightarrow{F}}$ and $u_0^{t,\overrightarrow{F}}$, and the transitions from these two terms come from t. We have several cases for t. If $t \rightarrow_{\mathrm{v}} t'$, then we still have $\langle s_i^{t',\overrightarrow{F}}\rangle\sigma_0^{\overrightarrow{E'}} \dots \sigma_j^{\overrightarrow{E'}} \ \mathcal{X}_{\mathcal{E}} \ \langle u_i^{t',\overrightarrow{F}}\rangle\delta_0^{\overrightarrow{E'}} \dots \delta_j^{\overrightarrow{E'}}$.

If $t = v$, then $\langle s_0^{v,\overrightarrow{F}}\rangle\sigma_0^{\overrightarrow{E'}} \dots \sigma_j^{\overrightarrow{E'}} = \langle F_0[\langle \boldsymbol{E}[v]\rangle]\rangle\sigma_0^{\overrightarrow{E'}} \dots \sigma_j^{\overrightarrow{E'}}$ and we also have $\langle u_0^{t',\overrightarrow{F}}\rangle\delta_0^{\overrightarrow{E'}} \dots \delta_j^{\overrightarrow{E'}} = \langle F_0[\langle \boldsymbol{E'}[v]\rangle]\rangle\delta_0^{\overrightarrow{E'}} \dots \delta_j^{\overrightarrow{E'}}$. It is easy to check that $\langle \boldsymbol{E}[v]\rangle$ and $\langle \boldsymbol{E'}[v]\rangle$ reduce to the same term $\langle t_0\{\lambda y.\langle E[y]\rangle/k\}\{v/x\}\rangle$, written t'. Then we have $\langle s_0^{v,\overrightarrow{F}}\rangle\sigma_0^{\overrightarrow{E'}} \dots \sigma_j^{\overrightarrow{E'}} \rightarrow_{\mathrm{v}} \langle F_0[t']\rangle\sigma_0^{\overrightarrow{E'}} \dots \sigma_j^{\overrightarrow{E'}}$, and also $\langle u_0^{t',\overrightarrow{F}}\rangle\delta_0^{\overrightarrow{E'}} \dots \delta_j^{\overrightarrow{E'}} \rightarrow_{\mathrm{v}} \langle F_0[t']\rangle\delta_0^{\overrightarrow{E'}} \dots \delta_j^{\overrightarrow{E'}}$; the two resulting terms are in the first set of $\mathcal{X}$. If $i > 0$, one can check that $\langle s_i^{v,\overrightarrow{F}}\rangle\sigma_0^{\overrightarrow{E'}} \dots \sigma_j^{\overrightarrow{E'}} \rightarrow_{\mathrm{v}} \langle s_{i-1}^{\langle F_0[t']\rangle,\overrightarrow{F'}}\rangle\sigma_0^{\overrightarrow{E'}} \dots \sigma_j^{\overrightarrow{E'}}$ and we also have $\langle u_i^{v,\overrightarrow{F}}\rangle\delta_0^{\overrightarrow{E'}} \dots \delta_j^{\overrightarrow{E'}} \rightarrow_{\mathrm{v}} \langle u_{i-1}^{\langle F_0[t']\rangle,\overrightarrow{F'}}\rangle\delta_0^{\overrightarrow{E'}} \dots \delta_j^{\overrightarrow{E'}}$, where $\overrightarrow{F'} = F_1 \dots F_i$ (the first context F_0 is removed from the sequence). We obtain terms that are in the second set of $\mathcal{X}$. In both cases, the resulting terms are in $\mathcal{X}$. The reasoning is the same if $t = k_l$ for some $0 \leq l \leq j$.

If $t = E'_{j+1}[\mathcal{S}k_{j+1}.t']$, then

$$\begin{array}{l}\langle s_0^{t,\overrightarrow{F}}\rangle\sigma_0^{\overrightarrow{E'}} \dots \sigma_j^{\overrightarrow{E'}} = \langle F_0[\langle \boldsymbol{E}[E'_{j+1}[\mathcal{S}k_{j+1}.t']]\rangle]\rangle\sigma_0^{\overrightarrow{E'}} \dots \sigma_j^{\overrightarrow{E'}} \\ \qquad\qquad\qquad\qquad\qquad\qquad \rightarrow_{\mathrm{v}} \langle F_0[\langle t'\rangle]\rangle\sigma_0^{\overrightarrow{E'}} \dots \sigma_j^{\overrightarrow{E'}}\sigma_{j+1}^{\overrightarrow{E'},E'_{j+1}}\end{array}$$

and

$$\langle u_0^{t,\overrightarrow{F}}\rangle\delta_0^{\overrightarrow{E'}}\dots\delta_j^{\overrightarrow{E'}} = \langle F_0[\langle \boldsymbol{E'}[E'_{j+1}[\mathcal{S}k_{j+1}.t']]\rangle]\rangle\delta_0^{\overrightarrow{E'}}\dots\delta_j^{\overrightarrow{E'}}$$
$$\rightarrow_{\mathrm{v}} \langle F_0[\langle t'\rangle]\rangle\delta_0^{\overrightarrow{E'}}\dots\delta_j^{\overrightarrow{E'}}\delta_{j+1}^{\overrightarrow{E'},E'_{j+1}}$$

therefore $\langle s_0^{t,\overrightarrow{F}}\rangle\sigma_0^{\overrightarrow{E'}}\dots\sigma_j^{\overrightarrow{E'}}$ and $\langle u_0^{t,\overrightarrow{F}}\rangle\delta_0^{\overrightarrow{E'}}\dots\delta_j^{\overrightarrow{E'}}$ reduce to terms of the form $\langle t''\rangle\sigma_0^{\overrightarrow{E'}}\dots\sigma_{j+1}^{\overrightarrow{E'}}$ and $\langle t''\rangle\delta_0^{\overrightarrow{E'}}\dots\delta_{j+1}^{\overrightarrow{E'}}$, that are in $\mathcal{X}_{\mathcal{E}}$. If $i>0$, then one can check that $\langle s_i^{t,\overrightarrow{F}}\rangle\sigma_0^{\overrightarrow{E'}}\dots\sigma_j^{\overrightarrow{E'}} \rightarrow_{\mathrm{v}} \langle s_{i-1}^{\langle F_0[t']\rangle,\overrightarrow{F'}}\rangle\sigma_0^{\overrightarrow{E'}}\dots\sigma_{j+1}^{\overrightarrow{E'}}$ and also $\langle u_i^{t,\overrightarrow{F}}\rangle\delta_0^{\overrightarrow{E'}}\dots\delta_j^{\overrightarrow{E'}} \rightarrow_{\mathrm{v}} \langle u_{i-1}^{\langle F_0[t']\rangle,\overrightarrow{F'}}\rangle\delta_0^{\overrightarrow{E'}}\dots\delta_{j+1}^{\overrightarrow{E'}}$, where $\overrightarrow{F'}=F_1\dots F_i$, so the resulting terms are in $\mathcal{X}_{\mathcal{E}}$.

If $t = F_{i+1}[k_l\ v]$ (with $1\le l\le j$), then

$$\langle s_0^{t,\overrightarrow{F}}\rangle\sigma_0^{\overrightarrow{E'}}\dots\sigma_j^{\overrightarrow{E'}} = \langle F_0[\langle \boldsymbol{E}[F_{i+1}[(\lambda y.\langle \boldsymbol{E}[E_l[y]]\rangle)\ v]]\rangle]\rangle\sigma_0^{\overrightarrow{E'}}\dots\sigma_j^{\overrightarrow{E'}}$$
$$\rightarrow_{\mathrm{v}} \langle F_0[\langle \boldsymbol{E}[F_{i+1}[\langle \boldsymbol{E}[E_l[v]]\rangle]]\rangle]\rangle\sigma_0^{\overrightarrow{E'}}\dots\sigma_j^{\overrightarrow{E'}} = \langle s_1^{E_l[v],\overrightarrow{F'}}\rangle\sigma_0^{\overrightarrow{E'}}\dots\sigma_j^{\overrightarrow{E'}}$$

and

$$\langle u_0^{t,\overrightarrow{F}}\rangle\delta_0^{\overrightarrow{E'}}\dots\delta_j^{\overrightarrow{E'}} = \langle F_0[\langle \boldsymbol{E'}[F_{i+1}[(\lambda y.\langle \boldsymbol{E'}[E_l[y]]\rangle)\ v]]\rangle]\rangle\delta_0^{\overrightarrow{E'}}\dots\delta_j^{\overrightarrow{E'}}$$
$$\rightarrow_{\mathrm{v}} \langle F_0[\langle \boldsymbol{E'}[F_{i+1}[\langle \boldsymbol{E'}[E_l[v]]\rangle]]\rangle]\rangle\delta_0^{\overrightarrow{E'}}\dots\delta_j^{\overrightarrow{E'}} = \langle u_1^{E_l[v],\overrightarrow{F'}}\rangle\delta_0^{\overrightarrow{E'}}\dots\delta_j^{\overrightarrow{E'}}$$

with $\overrightarrow{F'} = F_{i+1}, F_0, \dots F_i$, so the resulting terms are in $\mathcal{X}_{\mathcal{E}}$. If $i>0$, then $\langle s_0^{t,\overrightarrow{F}}\rangle\sigma_0^{\overrightarrow{E'}}\dots\sigma_j^{\overrightarrow{E'}} \rightarrow_{\mathrm{v}} \langle s_{i+1}^{E_l[v],\overrightarrow{F'}}\rangle\sigma_0^{\overrightarrow{E'}}\dots\sigma_j^{\overrightarrow{E'}}$, and we have also $\langle u_0^{t,\overrightarrow{F}}\rangle\delta_i^{\overrightarrow{E'}}\dots\delta_j^{\overrightarrow{E'}} \rightarrow_{\mathrm{v}} \langle u_{i+1}^{E_l[v],\overrightarrow{F'}}\rangle\delta_0^{\overrightarrow{E'}}\dots\delta_j^{\overrightarrow{E'}}$, so the resulting terms are in $\mathcal{X}_{\mathcal{E}}$, as required.

Finally, let $\lambda x.t_0\ \mathcal{E}\ \lambda x.t_1$ and $v_0\ \dot{\mathcal{E}}\ v_1$. It is easy to check that by definition of $\mathcal{E}$, the two terms $t_0\{v_0/x\}$ and $t_1\{v_1/x\}$ are of the form $t'\sigma_0^{\overrightarrow{E'}}\dots\sigma_i^{\overrightarrow{E}}$ and $t'\delta_0^{\overrightarrow{E'}}\dots\delta_i^{\overrightarrow{E}}$.

RESEARCH CENTRE
NANCY – GRAND EST

615 rue du Jardin Botanique
CS20101
54603 Villers-lès-Nancy Cedex

Publisher
Inria
Domaine de Voluceau - Rocquencourt
BP 105 - 78153 Le Chesnay Cedex
inria.fr

ISSN 0249-6399